

SUPER

Anno 2
n. 9 - Ottobre 1985

L. 7.500

Spedizione in
abbonamento
postale Gruppo III/70

9

SUPERSINC
la rivista per gli utenti
dei prodotti
SINCLAIR

sinc

Una pubblicazione della

J. soft EDITRICE

in collaborazione con

GRUPPO
EDITORIALE
JACKSON



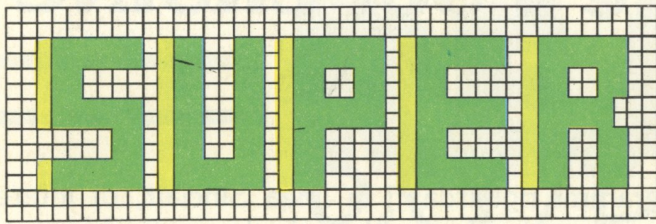
BOOMERANG WAR PER ZX81

QL: I MAGNIFICI QUATTRO

TOTORIDUTTORE



**UN NUOVO INSERTO:
RECENSIONI SOFTWARE**



6 **READ & WRITE**
LA POSTA DEI LETTORI

27 **REM:HW**
UN TASTO DOLENTE?
SOSTITUIAMOLO
di **Marcello Spero**

39 **TOTORIDUTTORE**
di **Massimo Serio**

9 **OPEN**
LA BANDA DEI
(MAGNIFICI?) QUATTRO
di **Marcello Spero**

31 **LOAD**

43 **Insert**
RECENSIONI
SOFTWARE

14 LA CULTURA **OS** INFORMATICA
...then
LISP: APRIAMO UNA PARENTESI
di **Lucio Bragagnolo**

31 **LABIRINTI**
di **Danny O'Mara**
trad. e adatt.
di **Marcello Spero**

59 **ZX CLUB**
I CLUB SINCLAIR

33 **SUPERNIM**
di **Francesca Giunta**

18 **RANDOMIZE**
L'ASSEMBLY ASSIEME
di **Marcello Spero**
Sesta parte

37 **BOOMERANG WAR**
di **Carlo Cappelli**

64 **PRINT**
RECENSIONI LIBRI

J. soft s.r.l.
**DIREZIONE, REDAZIONE,
AMMINISTRAZIONE**
Viale Restelli, 5
20124 MILANO
Tel. (02) 68.88.228-68.37.97
68.80.841/2/3

DIRETTORE RESPONSABILE:
Pietro Dell'Orco

COORDINAMENTO TECNICO:
Riccardo Paolillo

REDAZIONE:
Lucio Bragagnolo
Marcello Spero

**HANNO COLLABORATO A
QUESTO NUMERO**
Carlo Cappelli
Francesca Giunta
Massimo Serio

GRAFICA E IMPAGINAZIONE:
Margherita La Nocca
Ivana Rossi
Raffaella Toffolatti

FOTOCOPOSIZIONE:
Graphotek - Via Astesani, 16
Tel. 02/64.80.397
20161 MILANO

CONTABILITÀ:
Giulia Pedrazzini
Flavia Bonaiti

**AUTORIZZAZIONE ALLA
PUBBLICAZIONE:**
Tribunale di Milano n° 199
del 14.04.1984

STAMPA:
Provano
San Giuliano (MI)

PUBBLICITÀ
Concessionaria per l'Italia e
l'Estero J. Advertising s.r.l.
Viale Restelli, 5
20124 Milano
Tel. (02) 68.82.895-68.80.606-68.87.233
Tlx. 316213 REINA I

Concessionaria esclusiva per la
DIFFUSIONE in Italia e Estero:
SODIP - Via Zuretti, 25
20125 MILANO

Spedizione in abbonamento
postale Gruppo III/70
Prezzo della rivista L. 7.500
Numero arretrato L. 15.000
Abbonamento annuo: L. 82.500
(11 numeri con cassetta) L. 38.500
(11 numeri senza cassetta) per l'estero:
L. 110.000 (11 numeri con cassetta)
L. 50.000 (11 numeri senza cassetta)
i versamenti devono essere indirizzati a:
J. soft s.r.l.
Viale Restelli, 5
20124 MILANO

mediante emissione di assegno
bancario, vaglia o
utilizzando il c/c postale
n. 19445204.
Per i cambi di indirizzo indicare,
oltre al nuovo, anche l'indirizzo
precedente ed allegare alla
comunicazione l'importo di L. 500,
anche in francobolli
© TUTTI I DIRITTI DI
RIPRODUZIONE O TRADUZIONE
DEGLI ARTICOLI PUBBLICATI
SONO RISERVATI

GLI ARTICOLI TRADOTTI SONO
TRATTI DALLA RIVISTA
ZX COMPUTING - ARGUS
SPECIALIST
PUBLICATION LTD
1 GOLDEN SQUARE LONDON
W 1R3AB



Rivista associata
all'Unione
Stampa
Periodica
Italiana

Per collaborare a SUPERSINC

La rivista è interessata ad articoli e programmi riguardanti la linea Sinclair, di interesse generale, chiari ed esaurienti. Vi proponiamo una piccola "guida", che ha particolarmente lo scopo di rendere più accurata la stesura dei vostri manoscritti: seguendo i nostri consigli si accresceranno le probabilità che le vostre buone idee e i vostri programmi vengano pubblicati.

1 l'angolo superiore sinistro della prima pagina dovrà contenere: nome, cognome, indirizzo, numero telefonico, codice fiscale, data di spedizione, luogo e data di nascita.

2 l'angolo superiore destro della prima pagina dovrà contenere la marca e il tipo di computer al quale il lavoro si riferisce, unitamente ad eventuali espansioni di memoria o periferiche richieste.

3 il titolo dell'articolo, sottolineato, dovrà iniziare a circa due terzi in altezza della prima pagina.

4 le pagine seguenti potranno essere battute normalmente, con la condizione che l'angolo superiore destro contenga un'abbreviazione del titolo e del cognome, unitamente al numero di pagina. Per esempio, Horance Goes.../Brambilla/2.

5 il testo dell'articolo dovrà essere battuto a macchina con interlinea di uno spazio e mezzo, massimo due spazi; un margine di almeno un centimetro dovrà essere lasciato su entrambi i lati dello scritto.

6 dovranno essere usati fogli in formato UNI A4 (cm. 21 x 29,7) e il testo, scritto in caratteri maiuscoli e minuscoli, dovrà occupare una sola facciata del foglio.

7 nel caso il testo comprenda più fogli, questi dovranno essere uniti con un fermaglio o con un punto metallico aperto.

8 avendo intenzione di spedire più di un articolo, questi dovranno essere inviati separatamente insieme alla rispettiva copia su supporto magnetico.

9 programmi brevi (meno di 20 linee) potranno essere inseriti nel testo, mentre programmi più lunghi dovranno essere listati separatamente. È **ESSENZIALE** per noi disporre di una copia del programma registrata più volte su supporto magnetico, su entrambi i lati dello stesso. È preferibile usare nastri di buona qualità e di lunghezza non eccessiva; la cassetta o il disco dovranno essere etichettati con il nome dell'autore, il titolo dell'articolo, il computer interessato e soprat-

tutto le eventuali espansioni richieste. Come suggerimenti di programmazione si consiglia di usare, per esigenze di stampa listati, le istruzioni INK, PAPER, INVERSE, piuttosto che scrivere direttamente in INVERSE VIDEO. Un rapido controllo dei programmi per operare queste sostituzioni sarà da noi estremamente apprezzato.

10 per maggior chiarezza, all'interno dell'articolo è conveniente usare caratteri maiuscoli riferendosi a istruzioni BASIC (esempio RETURN, LIST, RND, PRINT ecc.). Se si desidera evidenziare una parola, è preferibile sottolinearla piuttosto che scriverla in carattere maiuscolo.

11 gli articoli ed i programmi potranno avere qualsiasi lunghezza: da una routine di una sola linea fino a programmi molto complessi.

12 volendo includere diapositive, queste dovranno avere formato 24x36, o 6x6.

13 non prenderemo in considerazione articoli che siano stati sottoposti ad altre case editrici.

14 il materiale non pubblicato non verrà restituito.

15 il compenso per la collaborazione prestata sarà commisurato alla complessità e all'interesse del testo e/o del programma (da un minimo di L. **50.000** a un massimo di L. **300.000**). Il pagamento verrà effettuato in caso di pubblicazione del lavoro.

Inviare idee e programmi a:
SUPERSINC
Viale Restelli, 5
20124 Milano

e saremo lietissimi di pubblicare i contributi migliori.

La Redazione

READ & WRITE

La posta
dei lettori

Tre cose sui microdrive

Spettabile redazione,
sarei interessato ad acquistare lo ZX Microdrive con relativa ZX Interface 1, ma prima di compiere il grande passo vorrei sapere tre cose: che tipo di software c'è in commercio? Sono di più i programmi su cassetta o su microdrive? Potrei trasferire i miei programmi dalle cassette ai microdrive?

**Massimo Perona
Camburzano (VC)**

Il software in commercio compatibile con i microdrive è prevalentemente professionale e hobbistico avanzato: si tratta per lo più di programmi di utilità (word processor, assembler, linguaggi alternativi, database...). Non ci risulta invece che esistano giochi già in vendita su cartuccia microdrive o microdrive compatibili, a parte quelli che vengono regalati con l'acquisto del cosiddetto "Expansion Pack", comprendente interfaccia 1 e microdrive, precisamente Ant Attack e Games Designer (il package comprende anche una cartuccia contenente il word processor Tasword II e il database Masterfile).

Detto questo si capisce facilmente come il rapporto tra programmi su cassetta e programmi su cartuccia sia enormemente a favore dei primi. È quasi sempre possibile trasferire su microdrive del software su cassetta, ma le difficoltà da superare in questo senso aumentano di pari passo col grado di sofisticazione delle varie protezioni usate, escludendo ovviamente i casi in cui i programmi siano dotati di un'apposita opzione di backup su cartuccia. I risultati in ogni caso sono eccezionali: le nostre copie di Manic Miner, tanto per fare un esempio, si caricano in una manciata di secondi.

Come giocare per sempre a Sabre Wulf

Per avere un numero infinito di vite in

"Sabre Wulf" occorre caricare il programma con MERGE "...". Digitando POKE 23756,1: CLEAR 65535 sarà possibile editare la linea di programma che avete caricato. Prima dell'istruzione PRINT USR 23424 inserite POKE 43575,255 (per un solo giocatore) o POKE 45520,255 (per due). Date ENTER, RUN, avviate il nastro e godetevi una partita senza fine!

**Paolo Birchieri - Cremona
Marco Sivori - Bolzaneto (GE)**

Il collaudo ha confermato l'efficacia del trucchetto. No comment...

Copiatori e copiandi

1) Voglio una risposta obbiettiva: è meglio lo Spectrum 48K o il Commodore 64?

2) Più di una volta mi è capitato, acquistando software presso rivenditori autorizzati, di vedermi offrire due versioni dello stesso: quella originale e quella che loro definiscono "personalizzata", identica alla prima (spudoratamente copiata), ma ad un prezzo di gran lunga inferiore (anche il 50%).

Inutile dirvi che ho sempre acquistato quest'ultima (non sarebbe umano rinunciare all'occasione); del resto voi cosa avreste fatto?

Se tuttavia il mio comportamento, sicuramente poco onesto, potrebbe essere giustificabile, non lo può essere quello di questi signori, causa principale della sproporzione esistente in Italia tra qualità e prezzo del software.

Più di una volta avete scritto che non pubblicherete mai copiatori, per non favorire il peraltro prolificante contrabbando di software. Ma siete realmente convinti di quanto affermate? Non ci vuole molto a capire che i "pirati" sono artisti nel loro campo, persone estremamente competenti: credete davvero che abbiano bisogno del programma pubblicato

da una rivista?

A conti fatti a rimmetterci sono quelli come me, autodidatti, con una povera infarinatura di BASIC, che non essendo in grado di duplicare in modo soddisfacente "Atic Atac" per la fidanzata sono costretti ad acquistarne un'altra copia!

Non pretendo di avere ragione, probabilmente ho torto marcio e la mia lettera sarà cestinata, ma per me è già un'enorme soddisfazione sapere che qualcuno l'avrà pur letta, prima di gettarla nel cestino.

Scusate lo sfogo.

Walter Mustoni - Arluno (MI)

Ci perdoni se trascuriamo la prima domanda (di cui si è già parlato e che darebbe luogo a interminabili discussioni accademiche, del tipo "sotto questo aspetto è meglio il tale computer, ma se consideriamo le capacità x, allora vale di più il tal altro") per fermarci più diffusamente sulla seconda, molto più problematica e importante per i programmatori e gli appassionati di informatica.

Innanzitutto vogliamo ribadire una cosa: copiare un programma per uso personale o, perché no, per mettere alla prova le proprie conoscenze di linguaggio macchina è un'azione assolutamente irreprensibile, che diventa ingiustificabile e da condannare quando, in qualunque forma, assuma scopo di lucro.

I personaggi da lei citati sono persone avide e senza scrupoli, ed ha perfettamente ragione lei quando afferma che è loro la principale responsabilità della "sproporzione tra qualità e prezzo del software" in Italia.

A questo punto, però, enunciato il principio, bisogna metterlo in pratica; noi al suo posto (ci perdoni la critica) non avremmo comprato neppure la copia a prezzo normale. Piuttosto, pirata per pirata, visto che lo spirito è forte, ma la carne è sempre debole, ci saremmo rivolti ai canali privati, dove è possibile trovare qualsiasi cosa (purtroppo) a prezzi stracciati, inferiori a quelli del più disonesto dei rivenditori.

Anche questa non è tuttavia la maniera migliore di agire, che resta quella di comprare software originale, se necessario anche spendendo qualche lira in più.

Si tratta, certo, di un discorso molto

difficile da sostenere di questi tempi e ancora più difficile da comprendere e fare proprio: ma è anche l'unica via che a gioco lungo possa tutelare, oltre che i diritti di chi scrive i programmi, anche quelli di chi, come lei e tanti altri, non possiede le conoscenze necessarie per proteggere, anche per il più innocente dei motivi, una cassetta. Se la pirateria non godesse del favore che ha presso gli appassionati, probabilmente per copiare "Atic Atac" basterebbe un qualsiasi SAVE "atic" CODE e il suo prezzo sarebbe molto più basso dell'attuale (basta pensare, se non altro, al costo supplementare dello studio di una protezione efficace).

Quanto ai canali privati di cui sopra, sono questi la principale ragione per cui non pubblichiamo programmi copiatori: pirati non si nasce, ma si diventa e, contrariamente a quello che lei crede, la maggior parte di essi non possiede affatto nozioni incredibili di programmazione: possiede semplicemente un programmetto di copia, di cui nel novantacinque per cento dei casi ignorano il perché del funzionamento. Presso questa rivista giungono giornalmente mucchi di annunci, compilati per lo più in grafia infantile, che annunciano la vendita di software a prezzi irrisori: tra di essi gli esperti sono una piccolissima minoranza. E diminuirebbero ancora di più quanto a percentuale, se le riviste come la nostra pubblicassero programmi di copia. Certamente, e disgraziatamente, ne fanno le spese tutti, anche chi non c'entra: questa situazione ci ricorda molto i tempi delle scuole elementari, quando alla fatidica domanda del maestro "chi è stato?" tutti tacevano senza che il colpevole avesse il coraggio di rispondere; inevitabilmente era tutta la classe a farne le spese. Probabilmente in questo campo dobbiamo ancora crescere un po', proprio come gli scolaretti: in Inghilterra esistono riviste dedicate esclusivamente (!) o quasi agli sprotettori, in cui tutti i giochi vengono sezionati fino all'impossibile e corredati delle più varie POKE relative a vite infinite, eliminazione di ostacoli, bonus di punti e via dicendo (c'è perfino chi si è messo ad aggiungere stanze a "Jet Set Willy"). È ovvio che con spiegazioni del genere sia possibile fare una copia del gioco: non si speri però di trovare su queste riviste un programma universale di copia. Noi preferiamo di gran lunga questa strada, più istruttiva e onesta delle altre, anche se più scomoda - per pigri e "furbi" - e impegnativa. Forse proprio in questo numero, a proposito, potrete trovare qualcosa del genere relativo ad "Ant Attack". Ma continueremo a rifiutarci di pubblicare programmi di copia. Anche se siamo i primi a non esserne contenti.

Avventurieri cercasi

Ci vorrebbe una rubrica che aiutasse i fanatici di avventure, in cui vi fossero

piccoli aiuti per superare i molti punti critici di questi giochi. Basterebbe una pagina...

Francesco Ghirotti - Ravenna

Purtroppo non abbiamo il tempo di giocarli (tutti)! Disponiamo però di spazio e volontà per pubblicare richieste di aiuto - ed eventuali risposte - di tutti i lettori. Perché non provare? In fondo finora si è parlato su queste pagine solo di "The Hobbit". È davvero un po' poco...

Personalizzare Ant Attack

Spettabile redazione, vi invio un metodo per sprotettare "Ant Attack", allo scopo di migliorarne la giocabilità.

Questo programma non è duplicabile con i copiatori più comuni in commercio. A quanto mi risulta, solo uno è in grado di farlo (di più, molti di più. N.d.R.), ma non ve ne dico il nome, anche per non alimentare il mercato abusivo di software.

Il sistema da me ideato serve per annullare l'azione del BREAK e per avere l'hard copy (stampa su carta, sempre. N.d.R.) del video: chi ha buona volontà e conosce l'inglese potrà anche tradurlo, personalizzarlo, complicarlo o semplificarlo, per salvarlo alla fine con le modifiche desiderate. Per fare ciò, ovviamente, bisogna possedere lo Spectrum 48K.

Eccovi ora la procedura:

- appena acceso il computer digitare PRINT 65536 - USR 7962. Se tutto funziona bene, dovrebbe apparire 41473 (sono i byte liberi di RAM). Prendere nota;

- procurarsi una cassetta vergine e salvare un programma in effetti inesistente, in questo modo:

SAVE "prova"CODE 23552,41400

(a questo punto apro una piccola parentesi per ricordare che Ant Attack è formato da due blocchi, di cui uno cortissimo e un altro senza header).

- digitare LOAD ""CODE e far leggere dal registratore solo l'header del programma inesistente: occorre cioè spegnere il registratore subito dopo la comparsa sullo schermo della scritta "Bytes: prova";

- togliere il nastro del programma inesistente dal registratore e inserire quello di Ant Attack;

- disinserire lo spinotto dalla presa EAR del registratore;

- azionare il registratore ed ascoltare fino a quando è passato il primo blocco di programma (quello cortissimo);

- inserire lo spinotto nella presa EAR e far partire il nastro: a caricamento completato apparirà un errore di tipo "Tape Loading Error". Premendo ENTER si potrà vedere il listato, e premendo RUN potrete giocare. Per interrompere il programma in una fase di gioco premete I

e, subito dopo, date BREAK: con tale sistema si può ottenere un hard copy del video. Se dà fastidio la scritta della linea 400, al fine di ottenere una stampa "pulita" digitare

400 PRINT AT 17,0; "scritta a piacere" ovviamente seguito da ENTER.

È possibile tradurre in italiano il programma, modificando le frasi, che appaiono nelle varie PRINT.

Al termine di tutte le operazioni digitate nuovamente

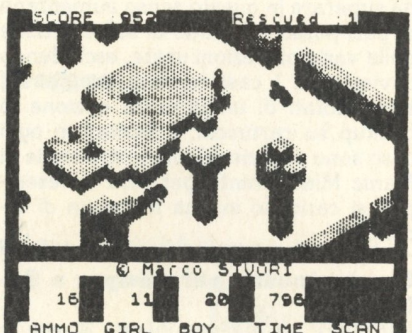
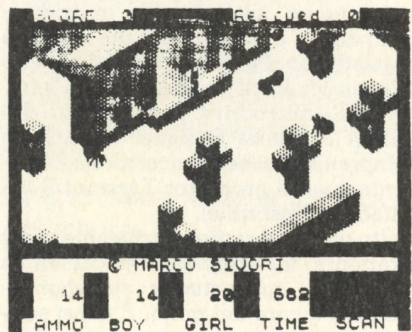
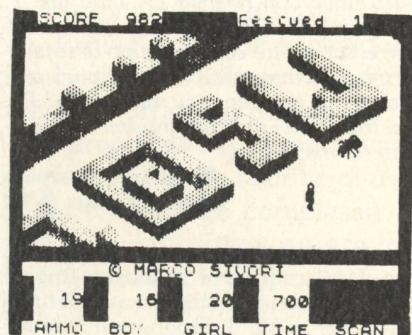
PRINT 65536 - USR 7962

che darà come risultato un valore x. Per salvare il tutto battete

SAVE "Ant"CODE 23552,41473-x+20 e... il gioco è fatto!

Marco Sivori - Bolzaneto (GE)

Il buon Marco, per convincerci della bontà del suo operato, allega alla lettera il listato del gioco e tre schermate ottenute con la stampante. Gli crediamo quindi sulla parola senza effettuare ulteriori collaudi: pubblichiamo inoltre le tre schermate, ben riuscite e - crediamo - per molti più che inedite.



Gli ultimi saranno i primi?

Giudicalo tu.

Anche se, in ordine di tempo, siamo gli ultimi a uscire con libri "dedicati" ai più diffusi home e personal computer, siamo convinti che il lettore attento ed esigente apprezzerà la qualità dei contenuti e la loro presentazione.

Abbiamo selezionato e tradotto i best-seller mondiali che accolgono quanto di più utile è stato scritto per il tuo computer.

Scegli a colpo sicuro!

Puoi ordinare direttamente i titoli presentati compilando ed inviando il coupon pubblicato oppure acquistarli presso i più qualificati computer shop e le migliori librerie.

CEDOLA DI ORDINAZIONE - LIBRI

Compilare e spedire in busta chiusa a:
J. soft - Viale Restelli 5 - 20124 Milano
tel. 02/6880841-6880842-6880843

Ordino i seguenti libri per un importo totale di L.
+ L. 2.000 come contributo fisso per spese di spedizione

Cod. Cod.
Cod. Cod.
Cod. Cod.

- Contanti allegati
- Assegno allegato n°
- Ho spedito l'importo a mezzo vaglia postale
- Ho versato l'importo sul CCP n° 19445204 intestato a J. soft - Milano
- Pagherò in contrassegno al postino al ricevimento dei volumi

Nome

Cognome

Via

CAP Città Prov.

Se richiesta fattura - codice fiscale

Data

J. soft: libri di qualità

Per Commodore 64

Giochi fantastici per il C 64
Cod. ASOC 001 L. 42.000
(inclusa cassetta)

Un libro pensato e realizzato per il divertimento dell'utente del C 64; contiene 19 giochi di vario genere tutti ampiamente commentati. L'attento studio dei programmi potrà essere vantaggioso per l'apprendimento delle tecniche di programmazione.

Il libro del C 64
Volume 1
Cod. ASOC 010 L. 24.000
(inclusa cassetta)
Il libro del C 64
Volume 2
Cod. ASOC 011 L. 24.000
(inclusa cassetta)

I due libri illustrano a fondo le possibilità del Commodore 64. Un compendio di utili consigli, e quant'altro è necessario per conoscere meglio il proprio calcolatore. Esempi pratici completano l'esposizione in modo chiaro ed esauriente.



Per ZX Spectrum

15 giochi grafici per ZX Spectrum
Cod. ASOC 002 L. 24.000
(inclusa cassetta)

Un fantastico volume con 15 entusiasmanti giochi grafici per il tuo ZX Spectrum. Tutti i programmi sono "pronti

Inoltre, i Paper Book: la raccolta dei programmi suddivisi per macchina di Paper Soft, il primo settimanale di software su carta per il tuo computer.

Paper Book - programmi per Apple II -
Cod. ASOC 005 L. 18.000
Paper Book - programmi per VIC 20 -
Cod. ASOC 003 L. 18.000
Paper Book - programmi per Commodore 64 -
Cod. ASOC 004 L. 18.000
Paper Book - programmi per Texas TI 99/4A -
Cod. ASOC 007 L. 18.000
Paper Book - programmi per ZX Spectrum -
Cod. ASOC 006 L. 18.000

J. soft s.r.l.

Viale Restelli, 5 - 20124 Milano - tel. 02/6880841-6880842-6880843



C 64: suono e grafica - Volume 1
Cod. ASOC 008 L. 24.000
(inclusa cassetta)
C 64: suono e grafica - Volume 2
Cod. ASOC 009 L. 24.000
(inclusa cassetta)

Due volumi che non possono mancare nella biblioteca dell'utente Commodore. Una miniera di idee e suggerimenti per la programmazione del tuo personal computer. Una vasta serie di programmi esemplificativi guidano il lettore al miglior sfruttamento del Commodore 64.

all'uso" e ampiamente commentati. I programmi sono scritti sfruttando appieno le capacità del calcolatore ed il loro studio può giovare a chiunque intenda affinare le proprie tecniche di programmazione.



In ogni volume, corredato di cassetta, oltre 40 programmi per il tuo computer: giochi grafici, utility, arcade, adventure una miscela esplosiva di software "pronto all'uso"!

Piccola guida per l'input dei programmi

I programmi pubblicati da SUPERSINC sono stati accuratamente provati e verificati: in questo modo speriamo di avere ridotto al minimo la possibilità di errori. Nel caso comunque un programma dovesse presentare malfunzionamenti, pubblicheremo le modifiche da approntare in uno dei numeri immediatamente successivi della rivista.

Per utilizzare i programmi pubblicati è sufficiente digitare i relativi listati sulla tastiera del calcolatore.

Nel caso dello ZX SPECTRUM, abbiamo previsto una semplice decodifica dei tasti da digitare nel modo grafico, per semplificare la comprensione dei listati.

Come è noto, lo SPECTRUM è provvisto di 2 serie di tasti grafici: una prima serie di 16 caratteri grafici predefiniti (i tasti numerici da 1 a 8 e gli stessi "shiftati") e una serie di caratteri definibili dall'utente nell'ambito di un programma (le lettere da A a U).

In entrambi i casi per ottenere i caratteri desiderati occorre entrare in modo grafico (cursore contrassegnato dalla lettera G lampeggiante), premendo contemporaneamente i tasti CAPS-SHIFT e 9.

Nei nostri listati i caratteri grafici predefiniti sono indicati da una Q seguita dal numero corrispondente al testo che occorre digitare, il tutto racchiuso tra due parentesi graffe.

Ad esempio {G4} significa che occorre digitare il tasto 4, con il cursore in modo grafico.

Analogamente la codifica SG, seguita da un numero da 1 a 8, significa che occorre digitare il relativo tasto numerico, premendo contemporaneamente il tasto CAPS-SHIFT.

Ad esempio quando, si trova la codifica {G2} occorre premere il tasto 2 contemporaneamente al tasto CAPS-SHIFT, ovviamente con il cursore in modo grafico. In entrambi i casi precedenti, quando un simbolo grafico deve essere digitato più volte, i caratteri G o SG della codifica sono preceduti da un numero che specifica quante volte va premuto il tasto grafico indicato.

Così ad esempio {8G5} significa che il tasto grafico 5 va digitato 8 volte e analogamente {4SG1} indica che il tasto grafico 1 e CAPS-SHIFT devono essere battuti 4 volte.

I caratteri grafici definibili (le lettere da A a U in modo grafico) hanno una codifica semplificata: la lettera corrispondente, sottolineata.

Quando in un listato viene presentata, ad esempio, una A sottolineata occorre entrare in modo grafico (al solito premendo contemporaneamente i tasti CAPS-SHIFT e 9) e quindi digitare semplicemente il tasto che contrassegna la lettera A.

Quando leggete	Premete	Vedrete
{G1}		
{G2}		
{G3}		
{G4}		
{G5}		
{G6}		
{G7}		
{G8}		
{SG1}		
{SG2}		
{SG3}		
{SG4}		
{SG5}		
{SG6}		
{SG7}		
{SG8}		

Se non siete già in modo G, entrateci schiacciando contemporaneamente CAPS SHIFT e 9

Se dovete uscire dal modo G, schiacciate 9

Quando leggete	Premete	Vedrete
<u>A</u>		Simbolo grafico definito nel programma in uso.
<u>B</u>		
<u>C</u>		
<u>D</u>		
<u>E</u>		
<u>F</u>		
<u>G</u>		
<u>H</u>		
<u>I</u>		
<u>J</u>		
<u>K</u>		
<u>L</u>		
<u>M</u>		
<u>N</u>		
<u>O</u>		
<u>P</u>		
<u>Q</u>		
<u>R</u>		
<u>S</u>		
<u>T</u>		
<u>U</u>		

Se non siete già in modo G, entrateci schiacciando contemporaneamente CAPS SHIFT e 9

Se dovete uscire dal modo G, schiacciate 9

OPEN

La banda dei (magnifici?) quattro

di **Marcello Spero**

Dopo le prime versioni del software Psion per il QL, caratterizzate da imperfezioni e bug più o meno evidenti, sembra che la versione 2.00 di tutti e quattro i programmi possa soddisfare anche l'utente più esigente. Vediamo fino a che punto è vera questa affermazione

A più di un anno dalla sua comparsa, da tempo ormai presente anche sul mercato italiano, il QL stenta ancora a trovare la sua strada. Certo, di problemi all'inizio ce ne sono stati, e molti. Il vero motivo della scarsa presa di questa macchina, che definire originale è davvero poco, sta probabilmente nell'alone di mistero che ancora la circonda, per merito di una politica, da parte dell'importatore ufficiale, a dir poco strana.

In questa atmosfera tipicamente inglese, resa ancor più inquietante dalla recente notizia di una parziale cessione dell'azienda da parte di Sir Clive (cosa succederà ora?), gettiamo uno sguardo illuminante (speriamo!) al software di base del QL nella versione rinnovata, uscita poco prima dell'estate.

Il cardine su cui ruotano le fortune del QL è costituito dai famosi quattro programmi di produzione Psion, forniti con la macchina. Finora, in effetti, si è trattato di un cardine piuttosto arrugginito, visto che i problemi più grossi sono nati proprio dalla scarsa compatibilità fra il QL ed il suo software.

Quella del pacchetto software Psion è una tempestosa storia, popolata di versioni 1.n, con n che va da 01 a chissà dove, tutte gravide di problemi più o meno pesanti, riassumibili nella considerazione che erano stati sviluppati prima che la macchina fosse disponibile, e quindi non erano sufficientemente "modellati" su di essa.

Scarsità cronica di memoria e frequenti cadute del sistema hanno popolato i pionieristici tentativi dei primi proprietari. Resasi finalmente disponibile la macchina nella sua versione definitiva (speriamo che sia davvero così) i programmatori della Psion si sono rimessi al lavoro per mondare le loro creature da tutte le macchie. È nata così la versione 2.00.

Finalmente scritti totalmente in linguaggio macchina, eccezion fatta per un brevissimo programma SuperBASIC di lancio e per le routine di installazione della stampante, i programmi occupano ora un volume di memoria considerevolmente inferiore, ed il loro funzionamento è decisamente più rapido e pronto. Anche il caricamento, prima molto lento a causa dei numerosi "overlay" (cioè sovrapposizione di nuove routine ad altre che, già utilizzate, non servono più) cui la Psion era ricorsa per limitare il più possibile l'occupazione di memoria, è adesso decisamente più veloce. In quindici secondi esatti si passa dalla pressione del tasto F1 o F2 (per l'autostart all'accensione della macchina) alla disponibilità del programma, qualunque dei quattro esso sia.

Anche gli accessi alla cartuccia del sistema, frequenti nelle passate versioni, non esistono più, fatta ovviamente eccezione per i file di help o, nel caso di Quill, per caricare in memoria i parametri di stampa relativi alla stampante che si vuole usare.

Resta qualche problema, ma in generale si tratta di cose di poco conto. Restano anche, purtroppo, le perplessità riguardo a Quill, il programma per l'elaborazione di testi. Ma procediamo con ordine, ed esaminiamo uno ad uno i quattro programmi che compongono il "cofanetto" fornito con il QL.

QUILL

È quello, fra i quattro, a risentire di più

delle migliorie apportate nel passaggio dalla versione 1 alla 2.

Il caricamento avviene, come abbiamo detto, in un batter d'occhio.

Altra novità importante, dovuta alla drastica riduzione della memoria occupata: ora non è più necessario inserire nel drive 2 una cartuccia perché il programma possa partire. Le versioni 1.n, infatti, necessitavano di un file, creato appunto sulla cartuccia nel drive 2 e in cui disporre di uno spazio di lavoro, per ovviare alla scarsità di memoria. Sembra impossibile, con 128K a disposizione, eppure era così. Il drive 2 può quindi tranquillamente essere vuoto. L'inserimento di una cartuccia verrà richiesto solo se il documento che state scrivendo supera all'incirca le tre pagine. Questo a meno che in memoria sia già presente qualcos'altro (altri programmi - in multitasking - od estensioni del SuperBASIC, tutt'e due cose poco compatibili con il software Psion; ma di questo parleremo un'altra volta), nel qual caso l'insufficiente spazio disponibile fa scattare il vecchio comportamento, con richiesta iniziale di seconda cartuccia.

Ora è in pratica possibile lavorare con Quill senza che in entrambi i drive ci sia alcuna cartuccia. Ovviamente, in questo caso l'help non sarà disponibile (l'help è una funzione di aiuto che richiama sullo schermo una spiegazione relativa alla situazione particolare in cui ci si trova al momento della richiesta; il suo testo è nella cartuccia di sistema del programma in uso), ma in compenso l'utente ha a disposizione due drive con cui compiere operazioni di backup (copia di file), merge (unione di più file in uno solo), ecc.

Anche il tempo impiegato per salvare un file è ora drasticamente diminuito.

Crash?

Malgrado i nostri ripetuti tentativi, pare impossibile mandare in "crash" il siste-

ma, come accadeva precedentemente, e questo vuol dire che i vecchi problemi sono stati risolti. Alcune cause di blocco, più nascoste e perciò più insidiose, esistono però ancora.

La prima, più frequente, riguarda il lavoro con documenti più lunghi di tre pagine, che quindi necessitano dell'apertura di un file di lavoro sulla seconda cartuccia. In condizioni normali l'utente quasi non si accorge di questa situazione; mentre il sistema compie le sue operazioni di ingresso ed uscita con il drive 2, è perfettamente possibile proseguire nell'uso del programma.

Man mano che il documento aumenta le sue dimensioni, ovviamente, il file si ingrandisce. Proprio questo è alla radice del possibile disastro. Specialmente se la cartuccia inserita nel drive 2 contiene anche altri file, cosa non infrequente visto il prezzo delle cartucce, può succedere che su di essa lo spazio si esaurisca. Nel momento in cui il sistema se ne accorge sullo schermo compare l'avvertimento, piuttosto ermetico, "errore di I/O - premere spazio per continuare".

L'utente che, ignaro, eseguisse tale direttiva si troverebbe allo stesso punto di prima, con la possibilità di proseguire all'infinito. Lampo di genio: e se inserisse nel drive 2 una nuova cartuccia, magari vuota? Mai più! Sarebbe la catastrofe! Il sistema ha bisogno assoluto del file già creato, completo o incompleto che esso sia.

Essendo qualsiasi operazione (persino il SAVE) impossibile, l'unico, amaro mezzo per recuperare il controllo sulla macchina è quello di cancellare penosamente parola dopo parola, fino a raggiungere una dimensione compatibile con lo spazio sulla cartuccia. Solo a quel punto si potrà salvare il documento, e quindi sostituire la cartuccia con una più "spaziosa".

Un altro mezzo per uscire dai guai è cancellare gli altri file eventualmente

presenti sulla cartuccia, ma è un metodo quasi certamente peggiore del primo.

Lo stesso tipo di messaggio viene dato in fase di salvataggio, se lo spazio in cartuccia non è sufficiente. In questo caso basta sostituire la cartuccia e il gioco è fatto. Occorre comunque tener presente che nel caso venga salvata una nuova versione di un documento con lo stesso nome della versione precedente (quindi in situazione di sovrascrittura) il sistema si accorgerà della mancanza di spazio solo dopo aver cancellato la vecchia versione, e ci si ritroverà perciò senza alcuna copia del file. Attenzione!

La seconda causa di crash è molto meno facile da sperimentare, ed è a nostra conoscenza unicamente perché la Psion lo ha gentilmente comunicato.

La nuova, velocissima routine di salvataggio ha infatti un limite costituzionale: non è in grado di salvare correttamente documenti composti da più di 32K (32767 caratteri), ossia circa 5500 parole. Tentando il salvataggio di file più voluminosi, la macchina si blocca invariabilmente.

La Psion dà la colpa di questo alla Sinclair, colpevole di aver prodotto un sistema operativo che non tiene fede alle specifiche. Alla Sinclair pare facciano finta di niente, brontolando che per questo prezzo non si può pretendere troppo...

Comunque sia, la Psion ha reso pubblica una "pezza" per rattoppare il problema.

Si tratta di una modifica che ripristina la vecchia e più lenta routine di salvataggio, presente sulle versioni 1.n.

In realtà si tratta di problemi di poco conto, facilmente prevenibili e prevedibili. Più gravi sono, secondo noi, le pecche che Quill ha sempre avuto, e ha tuttora, e che riguardano la sua struttura.

Manca infatti la possibilità di recupera-

re quanto si è cancellato, cosa molto comoda, visto che capita di frequente di andare "troppo in là" con la cancellazione. Una simile possibilità, presente in tutti i word processor di un certo livello, evita la sindrome da cancellazione, ovvero la paura attanagliante di perdere irrimediabilmente qualcosa di importante, e avrebbe potuto essere facilmente realizzata adoperando la freccia a destra, invece di utilizzarla per una ridondante cancellazione "a marcia indietro".

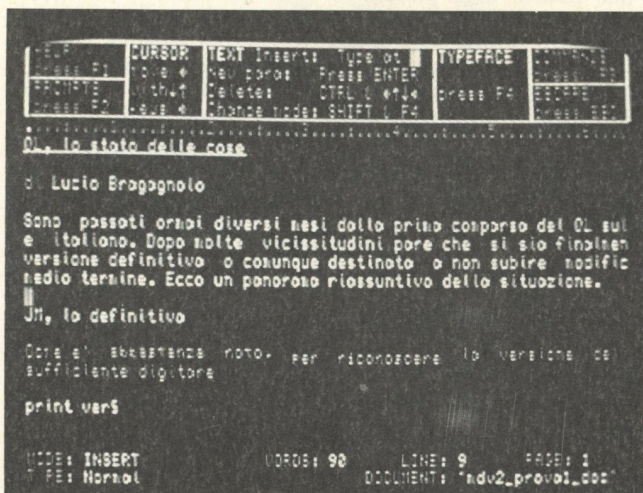
Altri grandi assenti sono un glossario, con possibilità di richiamare parole o frasi con la pressione di un unico tasto e la possibilità di usare più comandi uniti in programmi, richiamabili e conservabili (es. wpl Apple).

L'elenco potrebbe continuare, ma è forse più utile osservare come alla Psion non si siano troppo preoccupati di osservare cosa offriva il mercato al momento di scrivere il loro word processor.

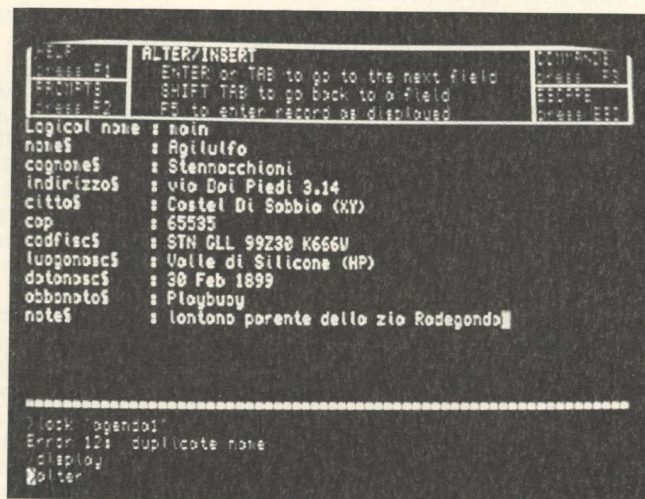
Pur con queste riserve, Quill resta un buon programma per l'elaborazione di testi, e oltretutto offre caratteristiche non disponibili in altri programmi simili, come la possibilità di avere pagine più ampie di quanto lo schermo consenta e margini variabili anche più volte all'interno dello stesso documento.

Un'impressione particolarmente positiva ce l'ha data il sistema di interfacciamento con la stampante, adattabile in un attimo a qualsiasi standard e capace di sfruttare a fondo le caratteristiche della stampante, collegata senza dover ricorrere a degli scomodi caratteri di controllo inseriti nel testo.

Traduzione automatica di caratteri, inserimento automatico di sequenze di controllo per un eventuale passaggio ad altro alfabeto o formato sono solo alcune delle caratteristiche del software di controllo per la stampante, caratteristiche presenti del resto anche in Archive.



L'inizio di questo articolo, realizzato con QUILL. Notare il grassetto e la sottolineatura, visibili anche sullo schermo.



ARCHIVER è un data base flessibile e di discreta potenza.

ARCHIVE

È un programma per la creazione, la modifica e l'uso di basi di dati.

Una base di dati è un file in cui le informazioni sono suddivise in record e campi. La situazione è analoga a quella di uno schedario in cui le informazioni sono suddivise in schede (i record nel nostro caso) e ciascuna scheda contiene le informazioni, ad esempio nome, indirizzo e numero di telefono, ordinatamente separate (i vari campi, nel nostro caso).

Una volta creato, un file può essere modificato o semplicemente utilizzato, cercando al suo interno, per mezzo di opportuni comandi, le informazioni desiderate.

Continuando nel nostro esempio dell'elenco di nomi, indirizzi e numeri telefonici, potremmo voler trovare la "scheda" (record) relativa a Mario Rossi.

Un apposito comando,

```
find "Mario Rossi"
```

lo farà. Volendo, potremmo vedere tutte le schede relative ai vari Rossi, o tutte quelle delle persone il cui numero telefonico inizia con 02.

Quando le operazioni di ricerca o di modifica diventano complesse, si sente la necessità di poter unire i vari comandi singoli in un programma, che oltretutto possa essere conservato e riutilizzato in futuro.

Questo con Archive è possibile, e bisogna dire che la sintassi del suo linguaggio è piuttosto avanzata, disponendo di strutture come

```
while
```

```
if...then...else
```

ed una completa proceduralità e ricorsività.

Non mancano funzioni anche molto complesse, che sfruttano piuttosto a fondo le possibilità del QL. Man mano vengono scritti i programmi si indentano addirittura da soli, cioè "rientrano" in corrispondenza dei cicli, delle if o delle procedure.

Non manca nemmeno la possibilità, utile a chi avesse intenzione di produrre software commerciale per Archive, di proteggere i programmi, cioè impedirne la lista. Stranamente manca invece la possibilità di proteggere i file, probabilmente perché il sistema operativo del QL non lo consente.

A proposito di software commerciale, c'è da dire che un programmatore esperto può, sfruttando a fondo le possibilità offerte dal linguaggio Archive, ottenere prodotti di livello elevato. La capacità di definire completamente la disposizione dei vari campi sullo schermo, e la grande flessibilità delle istruzioni di input, rendono infatti possibile la creazione di procedure di gestione di archivi anche molto complessi (è prevista la possibilità di manipolare contem-

OPEN

poraneamente più file) veramente "user friendly", cioè utilizzabili anche da chi non ha mai visto un computer.

Insomma, un programma che ci ha convinti e che meriterebbe ben più interesse di quanto abbia destato finora, forse perché è il meno immediato da utilizzare, vista la complessità legata alla sua grande potenza.

Unici nei sono l'eccessiva laconicità dei messaggi di errore e la possibilità di ritrovarsi, in particolari condizioni, intrappolati in strani cicli senza uscita, dovuti all'auto-ripetizione di particolari comandi.

Anche qui la routine di salvataggio è stata modificata per essere più veloce, e soffre dello stesso inconveniente di quella di Quill nel caso di file eccezionalmente voluminosi.

ABACUS

Notevole, pur senza le doti di Archive, è Abacus, versione aggiornata dell'ormai classico tabellone di calcolo, vale a dire un programma che consente di disporre i dati in celle all'interno di una matrice tipo "battaglia navale", con righe e colonne contrassegnate da lettere e numeri.

In determinate celle verranno poi collocate delle formule, che potranno in relazione i vari dati, in modo da produrre risultati.

Se ad esempio voglio che nella cella C10 compaia il risultato della somma delle celle da A1 ad A5, moltiplicate per il contenuto della cella B4, scriverò

```
(A1+A2+A3+A4+A5)*B4
```

o, meglio,

```
sum(A1:A5)*B4
```

Il risultato apparirà nella cella in cui abbiamo inserito la formula.

Nel momento in cui noi modificassimo il contenuto di una delle celle coinvolte nel calcolo il risultato verrebbe automaticamente aggiornato.

Le celle possono inoltre essere destinate a contenere testo, per rendere chiare le tabelle prodotte.

La possibilità di mettere in relazione un numero anche molto elevato di celle, di disporre di riferimenti alle celle di tipo assoluto o relativo (che si automodificano se la formula viene spostata in un'altra cella) e l'elevato numero di funzioni disponibili (tutte le principali funzioni algebriche, alcune funzioni temporali e persino funzioni finanziarie, come quella per calcolare il tasso di rendimento di un investimento, sulla base del capi-

tale impiegato e delle rendite in un certo lasso di tempo) rende possibili vere e proprie simulazioni, in cui la variazione sperimentale di uno o più valori si evidenzia immediatamente in una variazione dei risultati.

Interessante è la possibilità, molto comoda quando i valori in gioco diventano tanti, di riferirsi alle varie celle, non solo con lettere e numeri, ma con etichette create da noi.

Supponiamo ad esempio di aver chiamato le colonne dalla A alla L con i mesi dell'anno, e le righe 1 e 2, rispettivamente, costi e ricavi. Per riferirci alla cella B2 possiamo ora usare

```
ricavi.febbraio
```

con evidente guadagno in chiarezza.

Da notare è la possibilità di utilizzare vari tipi di incolonnamento per i dati nelle celle, come pure la possibilità di inserire automaticamente in un numero qualsiasi di celle dei prefissi (ad esempio \$) o suffissi (ad esempio %).

Ulteriori tocchi di raffinatezza sono la possibilità di suddividere lo schermo in due finestre separate, per lavorare contemporaneamente in due punti anche molto distanti della griglia, e di variare a piacere il tipo di unità in cui i dati sono espressi (decimale, intero, esponenziale, monetario, percentuale).

Come si è detto all'inizio, un buon programma, con molte qualità non comuni.

La disponibilità di memoria utilizzabile per i dati è ora di 23K, contro i 15 della precedente versione. Pur con questa maggior disponibilità di spazio non è tuttora possibile giungere al riempimento completo della griglia con dati numerici. Si tratta in realtà di un inconveniente da poco visto che normalmente più di metà griglia è occupata da testo o spazi vuoti.

EASEL

Se ad Archive spetta la palma di programma più potente, a Easel va senz'altro il titolo di programma più immediato da comprendere ed usare.

Easel è un programma per la generazione di grafici, partendo da dati numerici.

Può essere utilizzato a vari livelli. Al livello più semplice ci si limita ad introdurre uno dopo l'altro i dati, spostando il cursore attraverso i mesi dell'anno, o degli anni, a seconda del numero di dati a disposizione. La scala di lettura si modifica da sola, per adattarsi all'ordine di grandezza dei dati, ed il risultato appare come un istogramma. Già a questo

livello è possibile introdurre dati negativi, che provocheranno lo spostamento verso il centro dello schermo della linea di zero e la creazione di barre dirette verso il basso.

Un ulteriore livello di utilizzo è quello che prevede l'uso dei molti comandi a disposizione per modificare il formato, scegliendolo fra i vari tipi di barre e linee; è disponibile persino un diagramma a torta.

Addentrando maggiormente, è possibile utilizzare più serie di dati, ponendole in relazione in diagrammi complessi, in cui parte dei dati può essere espressa in forma di istogramma e parte in forma di linea, in senso orizzontale o verticale, variando a piacere colori e forma delle barre, delle linee, della griglia e della scala. Un testo di qualsiasi tipo può essere collocato in ogni punto e spostato in alta risoluzione, pixel per pixel.

Infine, dulcis in fundo, nel caso di uso contemporaneo di più serie di dati in diagrammi multipli compare automaticamente una chiave di lettura, che riporta i nomi da noi assegnati alle varie serie di dati ed i colori con cui essi sono rappresentati.

Per rendere possibili riproduzioni fotografiche dei diagrammi così realizzati è prevista la possibilità di eliminare ogni scritta non pertinente (indicazioni di comandi, ecc.). È comunque possibile la riproduzione a mezzo stampante grafica, e ne è previsto l'interfacciamento con i tipi più comuni.

Quello che colpisce di più in questo programma è il fatto che siano state previste anche le situazioni più strane e meno frequenti. Non solo, è del tutto impossibile mettere in difficoltà il sistema: in ogni caso il diagramma che otterremo

sarà pertinente.

Rispetto alle versioni precedenti anche qui i miglioramenti sono evidenti.

Innanzitutto la memoria disponibile, salita dai precedenti 8K a 12. Poi gli accessi continui alla cartuccia di sistema, che ora non esistono più. Ed ultima, ma non meno importante, la maggior velocità nel tracciare i grafici, particolarmente evidente nel caso dei diagrammi a torta.

Ci si rende subito conto, insomma, che si tratta di un programma ben studiato.

Peccato solo che sia così difficile realizzare copie su carta dei grafici, se non si dispone di una delle stampanti previste dal software di interfaccia, in questo caso non modificabile. L'unica possibilità è quella di effettuare uno "spool" dell'immagine video su cartuccia, e quindi ingegnarsi a creare un programma (in SuperBASIC?) che la sappia riprodurre correttamente tramite la stampante di cui si dispone.

Exchange

Comune a tutti e quattro i programmi è la possibilità di ottenere uno schermo a 40, 64 od 80 colonne, con un conseguente riadattamento di tutto quanto è visualizzato.

Sempre comune a tutti e quattro i programmi è anche un difetto, minimizzato rispetto alle precedenti versioni, ma pur sempre presente. Si tratta della difficoltà nello scambio di dati fra i vari programmi.

È molto comune, infatti, aver bisogno di rappresentare in forma grafica i risultati di un'elaborazione fatta con Abacus, o voler trasferire all'interno di un documento prodotto con Quill un indi-

rizzo conservato in un file di Archive. A questo scopo è previsto un apposito comando, Export, che produce file con formato universale. Questo purtroppo è vero soltanto in teoria, perché in pratica le limitazioni sono talmente tante da far desistere i più.

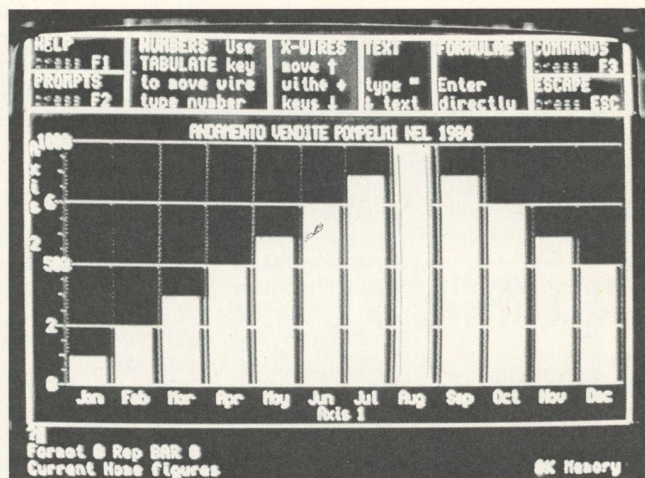
Estremamente allettante è l'annuncio della prossima commercializzazione di una cartuccia ROM, chiamata "Exchange", contenente l'unificazione dei quattro programmi in un superpacchetto unico, secondo quelli che sono gli ultimi orientamenti per il software di questo tipo. Tale cartuccia, che troverà posto non nell'apposita feritoria ROM, ma sul connettore di espansione RAM (viste le sue non indifferenti dimensioni) porrà finalmente fine ai problemi di trasferimento dati fra i singoli programmi.

In conclusione, si tratta di un pacchetto software estremamente allettante, seppure con le limitazioni descritte. La comparsa delle versioni 2.00 ha restituito professionalità a un software che non meritava di essere relegato fra i prodotti mal riusciti.

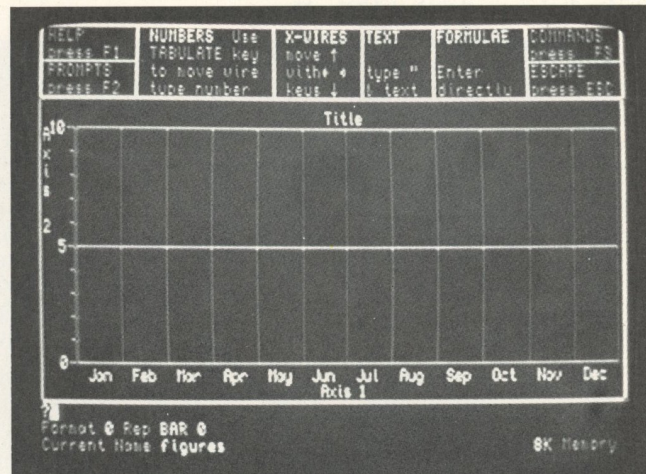
Peccato che la Psion non abbia approfittato per apportare qualche miglioria a Quill, "fratello povero" del quartetto.

A proposito: sapete che stanno per arrivare le versioni in italiano?

Concludiamo raccontandovi come, alla ricerca della fantomatica versione 2.00 e dopo aver avuto risposta negativa circa la sua disponibilità in Italia da parte dell'importatore ufficiale REBIT (eravamo in maggio), ci siamo imbattuti nell'ottimo importatore indipendente MI-PE.CO. di Ostia (Roma), che disponeva già delle macchine corredate delle nuove versioni, oltretutto a un prezzo estremamente interessante. Mah...



Un semplice grafico realizzato con EASEL, forse il migliore dei quattro programmi per immediatezza e facilità d'uso.



I comandi rimangono sempre gli stessi da programma a programma. Qui vediamo una tipica schermata dello spreadsheet ABACUS.

Resta tanto da fare:



l'Italia è al primo posto. GRAZIE ITALIA!

unicef

COMITATO ITALIANO

Piazza Marconi, 25 - 00144 Roma - Tel. 06/5924420-5917975/6 - c/c postale n. 26479006

UFFICIO DI MILANO - Via Solari 11 - 20144 - Tel.: 02/8370600 - **COMITATI REGIONALI** sull'elenco telefonico

...then

Lisp: apriamo una parentesi

di **Lucio Bragagnolo**

Una presentazione del Lisp, il linguaggio più usato nelle ricerche sull'intelligenza artificiale, disponibile su Spectrum e QL

"I linguaggi di programmazione si dividono in due categorie: il Lisp e gli altri".

Questa citazione risale al 1973, ed è oggi un po' meno vera di quanto non lo sia stata allora: negli ultimi anni sono comparsi alcuni nuovi linguaggi (come il Prolog), anch'essi orientati verso applicazioni concernenti l'intelligenza artificiale e i cosiddetti sistemi esperti, quelli che, in parole povere, vogliono racchiudere in sé le conoscenze e l'esperienza di un medico, o si piccano di saper trovare il petrolio molto più facilmente di qualunque geologo.

Questi linguaggi annoverano però tutti tra i loro principali antenati il Lisp, che da parte sua è tutt'altro che vecchio o antiquato; i suoi dialetti oggi disponibili sono molto avanzati, e decine di scienziati e ricercatori ne fanno quotidianamente il loro principale strumento di lavoro.

Vediamo dunque cosa ha portato alla nascita del Lisp e in che modo esso si differenzi dagli altri linguaggi di programmazione.

Da dove arriva il Lisp?

Senza tediare troppo con cenni storici non necessari, basterà dire che il Lisp affonda le sue radici più profonde e lontane nella logica matematica, e più in particolare nel lavoro di grandi matematici del periodo tra la prima e la seconda guerra mondiale, come Alonzo Church e Kurt Goedel, i cui contributi risultarono in seguito importantissimi, sia nel campo della logica che in quello della programmazione. Non pensate però che già nel 1930 esistesse un'attività programmatoria come quella odierna: in realtà l'inizio della programmazione "seria" va fatto risalire agli anni a cavallo della seconda guerra mondiale, mentre i primi linguaggi di alto livello (come appunto il Lisp, o l'Algol 68) sono nati negli anni '50. Il vero e proprio antenato del Lisp è l'IPL (Information-Processing Language, linguaggio per



l'elaborazione delle informazioni), sviluppato da Herbert A. Simon, J.C. Shaw e Allen Newell intorno al 1955. Il vero e proprio inventore del Lisp è John McCarty, del Massachusetts Institute of Technology - il famoso MIT -, che seppe riassumere il lavoro di chi lo aveva preceduto in un linguaggio di elaborazione di liste che si diffuse a macchia d'olio tra tutti i gruppi impegnati nelle ricerche sull'intelligenza artificiale, ed è tuttora attualissimo.

Non tutti avranno prestato attenzione alla presenza, nel precedente paragrafo, della parola "liste", accuratamente non menzionata in precedenza. In effetti le liste (vedremo più avanti cosa sono) costituiscono l'oggetto principale su cui lavora ogni tipo di Lisp, e il vero e proprio nucleo (sintetico e - vedremo anche questo - molto elegante) dell'intero

linguaggio, che riceve da esse anche il nome: Lisp sta infatti per LISt Processing, elaborazione di liste. La grafia esatta del nome dovrebbe essere LisP, ma l'uso corrente ha "minuscolizzato" la P.

Normalmente il Lisp è disponibile sui calcolatori sotto forma di linguaggio interprete (come il BASIC dello Spectrum e in generale di tutti i personal): è cioè possibile dare comandi diretti e ricevere subito la risposta appropriata, invece che dover forzatamente scrivere un programma completo prima di poterlo eseguire: è questo il caso della maggior parte dei Pascal, di solito disponibili come compilatori. Vedremo anzi come il Lisp sia uno dei linguaggi più liberi in assoluto per quanto riguarda il formato di presentazione dei programmi sullo schermo. L'uso di un interprete anziché

di un compilatore e le peculiarità del linguaggio stesso fanno sì che sia possibile costruire programmi istruzione dopo istruzione, gradualmente e risolvendo un problema per volta.

Tocchiamo con mano

Ora sediamoci davanti alla tastiera del nostro computer (lo Spectrum, ovviamente) e carichiamo l'interprete Lisp. Al termine del caricamento apparirà un "prompt" sotto forma di asterisco: grosso modo l'equivalente del cursore in BASIC: è lì per avvisarci che possiamo dare le nostre istruzioni al computer. Proviamo ora a digitare una cosa tutto sommato semplice come (plus 5 5) seguita da ENTER. Anche chi non ha mai sentito parlare del Lisp capirà che il risultato sarà la visualizzazione di 10, dopo di che l'interprete mostrerà nuovamente l'asterisco per darci modo di effettuare successive operazioni. Lo schermo dovrebbe presentarsi così:

```
*(plus 5 5)
10
*
```

Cos'è successo? L'interprete ha *letto* il nostro comando, lo ha *valutato* e ne ha *stampato* il risultato, prima di mostrarsi pronto a leggere qualcos'altro. In termini più corretti ha eseguito un *read-eval-print loop*, ossia un cosiddetto ciclo di lettura-valutazione-stampa. Ogni volta che viene battuta un'istruzione l'interprete ripete questo ciclo, e il sistema per calcolare più istruzioni è quello, come si diceva più sopra, di farle risolvere all'interprete una alla volta (esiste comunque la possibilità di scrivere programmi in forma più convenzionale). Che differenza rispetto a un, per dire, programma BASIC, in cui la soluzione di un problema dipende da quella del precedente, e in cui tutti i problemi devono essere presentati in un ordine ben definito! In BASIC basta un piccolo errore per bloccare l'intero programma, mentre in Lisp un errore blocca semplicemente un'istruzione... e questa facilità di lavoro è uno dei fattori che ha concorso alla diffusione del Lisp.

Esaminiamo ora un altro aspetto di ciò che abbiamo fatto: la notazione usata per dire all'interprete di aggiungere i numeri 5 e 5. In Lisp il simbolo di operazione precede sempre i suoi operandi, come previsto dalle regole della *notazione polacca*. Quest'ultima fu inventata dal logico polacco Jan Lukasiewicz, con non eccessiva fortuna. Infatti tutti si ricordano del suo opposto, la notazione polacca inversa, che i fedelissimi di IF... THEN si ricorderanno essere prerogativa del Forth.

Ora, per impratichirci con la notazione polacca, proviamo a digitare qualcosa di più complicato: l'equivalente di 27/(12+(7-4)*5).

```
*(div 27 (plus 12 (times 5 (diff 7 4))))
```

Riuscite a capire cosa succede? Forse

LA CULTURA INFORMATICA

...then

è meglio indentare un po' il listato:

```
*(dive 27
  (plus 12
    (times 5
      (diff 7 4))))
```

Partendo dalla prima operazione da eseguire, l'interprete calcola 7-4 (diff è la sottrazione: l'avevate capito, vero?); il risultato viene moltiplicato (times) per 5; il tutto viene sommato a 12 e successivamente diviso (div) per 27. Il risultato, per la cronaca, è 1. Sempre per la cronaca, div è una divisione tra interi, senza decimali: (div 14 3) è equivalente a 14/3, ma il suo risultato sarà 4 e non 4.66666667 come in BASIC. Guardando cosa abbiamo fatto, possiamo riflettere su un paio di cose: la meno importante è la profusione di parentesi, per alcuni un fastidioso difetto del Lisp che porta a commettere facilmente errori di sintassi (qualcuno ha malignamente supposto che Lisp stia per "Lots of Irritating Single Parentheses", mucchi di irritanti parentesi spaiate); la seconda è la composizione degli enunciati e quindi delle liste.

La lista dal cuore atomico

Le strutture su cui agisce l'interprete sono due: le liste, appunto, e gli atomi. Un atomo è... un atomo, un nome a cui può o meno essere associato un valore. Esempi tipici di atomi possono essere pippo, 1555, nome__di__atomo, a, 37. L'atomo è, come dice il nome stesso, indivisibile. Una lista è una collezione di oggetti racchiusi tra parentesi e messi in ordine: oggetti che possono essere atomi o... altre liste. Nell'esempio fatto sopra la lista (diff 7 4) è composta di tre elementi, esattamente come la lista (times 5 (diff 7 4)); il terzo elemento della lista è - chiamiamola così - la sottolista (diff 7 4). Anche la lista (plus 12 (times 5 (diff 7 4))), attenzione al conto delle parentesi!, è composta da tre elementi: plus, 12, e la sottolista (times 5 (diff 7 4)), la quale al suo interno contiene la sottosottolista (diff 7 4). La lista (div 27 (plus 12 (times 5 (diff 7 4)))) è composta, guarda un po', da tre elementi, di cui il terzo... chiaro per tutti? Per sgombrare il campo da ogni possibile equivoco diciamo che le liste non sono composte obbligatoriamente da tre elementi: (primo secondo contorno formaggio frutta dolce) è una lista di sei atomi valida a tutti gli effetti. Anche le istruzioni sono liste, come abbiamo potuto vedere, ed è molto importante che da un'istruzione possano scaturirne fuori altre, proprio come scatole cinesi! Dobbiamo infine parlare della lista vuo-

ta; istintivamente saremmo portati a contraddistinguerla come (), il che va benissimo per tutti i dialetti esistenti, i quali usano però correntemente al posto di questa notazione un sinonimo: "nil". Nil è un po' lo zero del Lisp, ed è sia un atomo che una lista. Il valore dell'atomo "pippo" è variabile a nostro piacimento; quello dell'atomo "456" è ovviamente 456, e non è modificabile; il valore di nil, anch'esso immutabile, è ..nil. Esiste un altro atomo particolare con valore uguale a sé stesso, l'atomo "t", abbreviazione di True (vero), di cui vedremo più avanti una delle applicazioni più abituali.

Come è possibile assegnare un valore a un atomo? Attraverso l'istruzione "setq". (setq quick 18), (setq quick (plus 6 6 6)) e (setq quick (times 9 2)) sono tre liste di uguale significato, che assegnano all'atomo "quick", come se fosse una variabile BASIC, il valore 18. La differenza nei riguardi del BASIC sta nel fatto che il valore di un atomo può essere anche una lista: basta digitare (setq quack (quote (diff 20 3))). La parola quote (sostituibile nella maggior parte dei sistemi con un semplice apostrofo) fa sì che ciò che la segue venga preso così com'è senza essere valutato. Risiediamoci alla tastiera:

```
*(setq quick (times 9 2))
18
*(setq quack (quote (diff 20 3)))
(diff 20 3)
```

Ricordiamo che le righe senza asterisco sono le risposte dell'interprete. Continuiamo:

```
*quick
18
*quack
(diff 20 3)
```

Capita la differenza? È quasi la stessa che intercorre in BASIC tra la variabile quick con valore 18 e la stringa quack\$ uguale a "17". In BASIC per trasformare quack\$ in un numero si userebbe VAL; in Lisp il comando "eval", che sta per "evaluate", valuta.

```
*(eval quack)
17
```

Lavorare sulle liste: car e cdr

Chiariti i concetti di lista e di atomo possiamo vedere un paio di istruzioni che ci consentono di operare su parte degli elementi di una lista, vale a dire "car" e "cdr". Car estrae il primo elemento di una lista, e cdr estrae tutta la

lista escluso il primo elemento. Per chiarire le cose, il car di (diff 20 3) è "diff"; (20 3) è il cdr. Attenzione, però: così come le abbiamo enunciate, queste operazioni, almeno sullo Spectrum, non sono eseguibili, in quanto l'interprete, non vedendo la parola "quote" davanti alla lista, la valuterà, generando un errore; infatti (car (17)) non è un'operazione eseguibile. Occorre comportarsi in questa maniera:

```
*(car (quote (diff 20 3)))
diff
*(cdr (quote (diff 20 3)))
(20 3)
```

Una domandina subdola, ora: qual è il car di ((pippo pluto)(paperino))? Chi risponde (pippo) farà bene a rileggersi l'articolo. La risposta esatta è infatti - verificiamo sullo Spectrum - questa:

```
*(car (quote ((pippo pluto)(paperino)))
(pippo pluto)
```

in quanto il primo elemento della lista è appunto la sottolista (pippo pluto). Pippo è il car di (pippo pluto), ovvero

```
*(car (car (quote ((pippo pluto)
(paperino))))))
```

Queste acrobazie sintattiche sono ovviamente eseguibili anche con cdr, o con car e cdr insieme. Il car di (nil), cioè di una lista contenente nil, è nil - esattamente come il cdr di (nil) - e il car (o cdr) di nil, visto come semplice atomo, provoca una segnalazione di errore, se non in certi casi un blocco del sistema, un po' come la divisione per zero in BASIC.

Giocando con car, cdr e le altre numerose funzioni disponibili all'interno di un interprete Lisp è dunque possibile manipolare a piacimento qualunque lista. Per ragioni di spazio non si può parlare di tutto: ci limitiamo ad accennare all'esistenza di altre due funzioni, reverse e cons. La prima inverte l'ordine degli elementi in una stringa, la seconda "fonde" assieme due liste diverse:

```
*(reverse (quote ((avanti)(indietro)))
(indietro)(avanti))
*(setq numero 2)
2
*(cons numero (quote (volte)))
(2 volte)
```

Se al posto di "(cons numero)" avessimo messo "(cons (quote (numero)))", l'atomo "numero" non sarebbe stato valutato e il risultato sarebbe stato (numero volte).

Funzioni e ricorsività

Come abbiamo visto, lavorando sulle liste è possibile realizzare un po' di tutto; ma la grande, vera potenza del Lisp sta nella possibilità dell'utente di definire funzioni e nella possibilità di farlo ricorsivamente, ossia utilizzando una funzione che viene definita tramite... se stessa. Cominciamo a esaminare il primo aspetto.

È chiaro che, essendo le parole chiave trattate in Lisp alla stregua di oggetti contenuti nelle liste, deve essere piuttosto facile costruire, manipolando le liste, funzioni di nostra invenzione, che si comportino esattamente come parole chiave, anzi: le "parole chiave" sono per la stragrande maggioranza funzioni che avrebbero potuto agevolmente essere definite dall'utente e che solo per sua maggiore comodità sono state incluse direttamente nell'interprete. Come si definisce una funzione? Dando all'interprete nome, argomento e parametri necessari. L'istruzione da usare è "de". Ora definiremo alla tastiera la funzione "quadrato", che consente di elevare al quadrato un numero intero qualunque. Fate attenzione:

```
*(de quadrato (argomento)(times argo-
mento argomento))
quadrato
```

dove "quadrato" è il nome della nuova funzione, (argomento) è l'argomento su cui va compiuta l'operazione e (times argomento argomento) è l'operazione che va compiuta. Ora "quadrato" è una funzione come car, cdr, reverse e tutte le altre. Proviamo a usarla:

```
*(quadrato 5)
25
```

Visto? Naturalmente, le nuove funzioni, come accade nel Forth, possono essere usate per definirne altre ancora, e così via, con limiti dettati solamente dalla memoria disponibile e dalle capacità del programmatore.

Esistono numerose altre funzioni del Lisp di cui si potrebbe parlare, ma la nostra breve carrellata ci consentirà solamente di accennare ad alcune di esse, che ci torneranno utili per illustrare le capacità ricorsive di questo linguaggio.

Consideriamo ad esempio la nozione di fattoriale: il risultato della moltiplicazione di un intero con tutti i numeri (interi) che lo precedono, escluso naturalmente lo zero. La notazione di fattoriale prescrive l'utilizzo di un punto esclamativo da sistemare a destra del numero "fat-

toriale". Il fattoriale di 5 è perciò 5!, l'equivalente di $5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 120$. Proviamo a scrivere un programmino BASIC che calcola il fattoriale di un numero:

```
10 INPUT cifra
20 LET totale=1
30 FOR f=2 TO cifra
40 LET totale=totale*cifra
50 NEXT f
60 PRINT totale
```

Il Lisp ragiona in tutt'altro modo, appunto ricorsivamente. Vediamo il calcolo del fattoriale scritto in Lisp e opportunamente indentato:

```
*(de fattoriale (cifra)
(cond ((zerop cifra) 1)
(t (times cifra (fattoriale
(sub1 cifra))))))
fattoriale
```

Siete inorriditi? State calmi, ora chiariamo tutto, riga per riga.

La prima riga ospita la definizione della nuova funzione e il suo argomento; la seconda e la terza le operazioni da eseguire, che condensano in uno spazio minimo le operazioni che occupano tutto il programmino BASIC, con l'eccezione delle linee 10 e 60. La funzione cond è l'equivalente BASIC di IF...THEN: l'interprete scorrerà il programma a partire da cond, eseguendo solo le linee che verificano una condizione data e ignorando le altre. Zerop esamina cifra e guarda se contiene o meno zero. Se ciò accade, cifra assume valore 1, altrimenti il programma passa alla linea successiva, la quale fa sempre capo a cond. In quest'ultima la presenza iniziale dell'atomo "t" fa sì che essa venga eseguita forzatamente (o l'interprete, saltandola, andrebbe a cercare una linea successiva che non esiste), calcolando il prodotto del numero inserito per il fattoriale (che così si autodefinisce, vedete?) del numero stesso decrementato di 1 tramite la funzione sub1. In parole povere, l'interprete esegue una specie di conto alla rovescia, ragionando in questo modo: "Dunque, devo calcolare questa nuova funzione chiamata fattoriale della cifra che mi è stata data. Io so solamente che il fattoriale di 0 è uguale a 1, come mi è stato detto nella seconda riga. Allora sarà meglio che, intanto, moltiplichi il numero per il fattoriale del numero stesso diminuito di 1; il nuovo numero non è zero, per cui è meglio che lo moltiplichi, intanto, per il fattoriale del numero stesso diminuito di 1; il nuovo numero... è zero. Finalmente! Posso fermarmi alla seconda riga, moltiplicando tutto per 1 e restituendo il totale di quanto ho calcolato finora". Insomma, l'interprete, incapace di calcolare il fattoriale - poniamo - di 3, "prende tempo", cercando di trovare il fattoriale di 3-1, il fattoriale di 2-1, il fattoriale di..., diminuendo ogni volta l'argomento della funzione (la

quale si richiama da sola ogni volta, automodificando il proprio argomento e così via) fino a trovare il caso che lui è in grado di padroneggiare, cioè quello, in questo caso, in cui l'argomento stesso è uguale a 0. In BASIC si può ragionare in questa maniera solo disponendo della possibilità di definire procedure (come nel caso del QL): sullo Spectrum l'unico programma che, come concetto, si avvicina a ciò che abbiamo appena visto è riassumibile in un obbrobrio quale

10 GO SUB 10

capace di provocare un "Out of memory" in pochissimo tempo.

Per ora è tutto...

Prima di chiudere, una parola sugli interpreti Lisp disponibili per i computer Sinclair: per lo Spectrum esiste lo SpecLISP 1.2 della Serious Software, il cui indirizzo è 7 Woodside Road - Bickley, Kent BR1 2ES. Si tratta di un ottimo interprete, considerato anche il pochissimo spazio che occupa in memoria, con la possibilità di poter chiamare, tra le altre cose, eventuali routine in linguaggio macchina risidenti sopra RAM-TOP. Con lo SpecLISP sono stati celebrati gli esempi riportati in questo articolo. Per quanto concerne il QL, il Lisp è prodotto (per ora: sono stati annunciati vari interpreti e compilatori da parte di altre case) dalla Metacomco, avente come indirizzo 26 Portland Square - Bristol, BS2 8RZ. Il programma comprende anche un buon screen editor, e può fare uso senza problemi delle capacità grafiche e di gestione finestre del QL. Con queste notizie si chiude la nostra superficialissima e breve panoramica su questo affascinante linguaggio: ci sarebbero ancora moltissime cose da dire, ma pensiamo che queste pagine siano sufficienti per dare anche ai meno interessati all'argomento un'idea di cosa sia il Lisp. Non è escluso che, se il tema riscuoterà l'approvazione dei lettori, ci si torni sopra in maniera più approfondita, magari con un vero e proprio corso, accompagnato da qualche puntata nel mondo dell'intelligenza artificiale, di cui il Lisp rimane sempre l'alfiere indiscusso.

Bibliografia:

Laurent Siklossy - Let's Talk Lisp - ed. Prentice-Hall - 1976
 Ken Tracton - Programmer's Guide to LISP - ed. Tab Books - 1980
 Robert Wilensky - LISPcraft - W.W. Norton & Company - 1984
 Douglas H. Hofstadter - Le Scienze (rubrica "Temi metamagici") - articoli pubblicati sui numeri 176/178 datati Aprile/Maggio/Giugno 1983
 Giuseppina Gini/Maria Gini/Giovanni Guida - Lisp/linguaggio e metodologia di programmazione - CLUP - 1981.

Quest'ultimo libro può essere richiesto alla J.soft, con codice ACLK003, al prezzo di Lit. 11700.

Per ricevere le riviste arretrate di SUPERSINC con cassetta compilate e spedite il coupon qui sotto!

SUPERSINC

CEDEOLA DI ORDINAZIONE RIVISTE ARRETRATE CON CASSETTA ALLEGATA da compilare e spedire in busta chiusa a J.soft - Viale Restelli, 5 - 20124 Milano - Tel. 6888228

Vogliate inviarmi i numeri di SUPERSINC con cassetta

Al prezzo di L. 15.000

Contanti allegati Assegno allegato n°

Ho spedito l'importo a mezzo vaglia postale

Ho versato l'importo sul cc. n° 19445204 intestato a J. soft - Milano

Pagherò in contrassegno al postino al ricevimento

BUONO D'ORDINE PER RIVISTE ARRETRATE
 Anche se l'ordine riguardasse la sola cassetta questa verrà comunque inviata insieme alla rivista al prezzo qui indicato.

Nome

Cognome

Via

CittàC.A.P.Prov.

Se richiesta fattura:

Cod. F. e P. Iva

Data

Firma

Per i residenti all'estero - pagamento anticipato (vaglia o versamento su ns. ccp)

RANDOMIZE

L'Assembly assieme

di **Marcello Spero**
Sesta parte

Movimentiamo un po' lo schermo del nostro Spectrum, ovvero come scrollare a piacimento

Digerito il "matteone" della scorsa puntata? Non preoccupatevi: questa volta ci terremo più "leggeri".

Ciò che vedremo è infatti una serie di routine piuttosto brevi e dalla struttura di cristallina chiarezza.

Fanno tutte la stessa cosa; si tratta infatti di routine per lo "scrolling" laterale dello schermo. Ciascuna esegue però questa operazione in modo diverso. Troviamo infatti lo scroll per pixel, per mezzo byte e per byte, con o senza attributi..., ma procediamo con ordine.

Lo scrolling

Innanzitutto, per i meno addentro a queste cose: cos'è lo "scrolling"?

Si tratta semplicemente dello spostamento dell'immagine video.

Ovviamente, questo può avvenire in quattro diverse direzioni: alto, basso, destra e sinistra. Delle prime due non ci occuperemo, visto che non ci serviranno e che la complessità di una loro realizzazione sarebbe notevole.

D'altronde, la possibilità di ottenere lo "scrolling" verso l'alto di un carattere per volta esiste già nello Spectrum. Normalmente questo movimento avviene automaticamente quando lo schermo è pieno, per arrestarsi periodicamente con la richiesta "Scroll?". Per ottenere uno scrolling continuo, senza l'apparizione della richiesta, basta dare continuamente alla variabile di sistema SCR CT (locazione 23692) valori maggiori di 1 (normalmente si usa un periodo POKE 23692,255).

Per quel che ci riguarda, invece, ci interesseremo degli spostamenti laterali.

Si tratta, dicevamo, di operazioni concettualmente più semplici rispetto agli spostamenti verticali, per un motivo ben preciso: la configurazione dell'area video.

Nel senso verticale, infatti, la disposizione in memoria delle varie linee di 8 pixel (cioè byte) che compongono ciascun

carattere non rispecchia affatto la loro posizione sullo schermo televisivo. Chiamando 1.1 la prima linea della prima riga di caratteri, 1.8 l'ultima della prima, 2.1 la prima della seconda riga e così via, mentre sullo schermo abbiamo, ovviamente:

1.1
1.2
1.3

1.8
2.1
2.2

e avanti di questo passo, in memoria

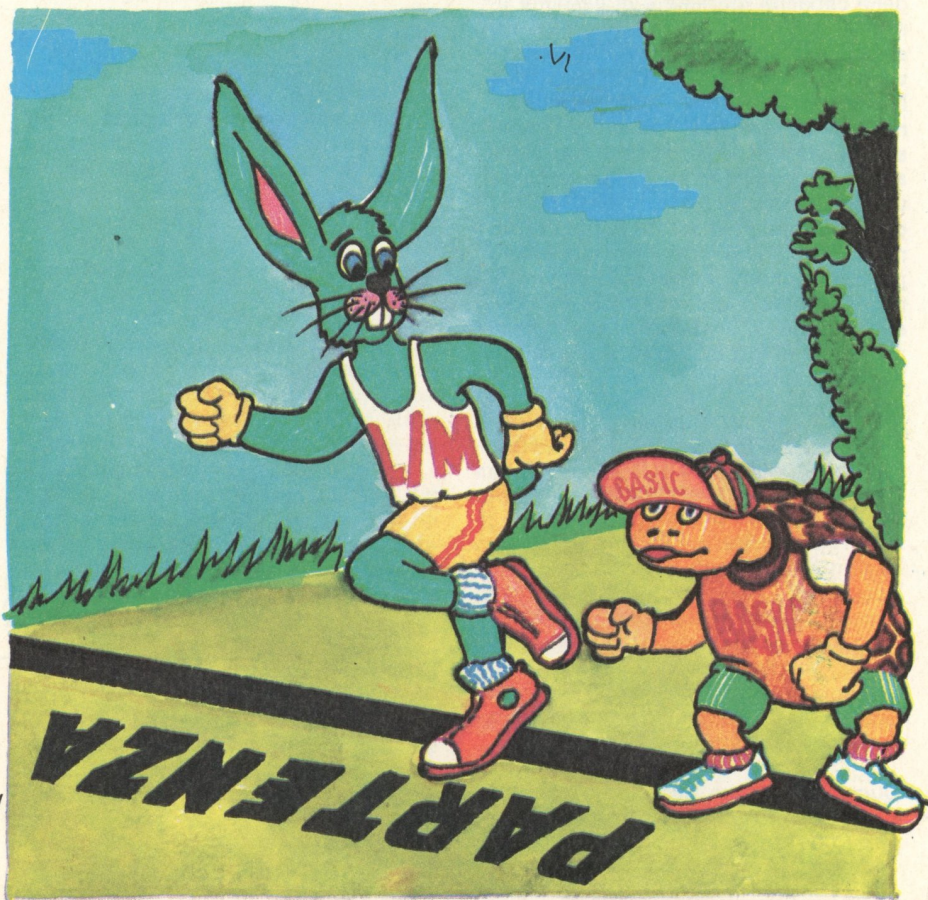
troviamo:

1.1
2.1
3.1

8.1
1.2
2.2
3.2

8.2
1.3

e così via. Dapprima, cioè, la prima linea delle prime otto righe di caratteri, poi la seconda, la terza, ecc., fino all'ot-



tava ed ultima. La medesima disposizione si ripete quindi con la seconda serie di otto righe e con la terza (il numero totale di righe presenti sullo schermo è infatti 24, contando anche le ultime due riservate ai comandi).

In senso orizzontale le cose sono invece molto più semplici, perché la successione è quella logica. Infatti, mentre al di sotto del quarto byte del carattere 6,8 (il nono della settima riga) non ne troveremo il quinto, alla sua destra ci sarà certamente il quarto byte del carattere 6,9 (il decimo della settima riga). Possiamo quindi dire che le varie linee (in tutto sono 24x8=192) sono rimescolate, ma le colonne non lo sono.

Con l'ausilio del listato 1 potrete comprendere meglio la struttura dell'area video. Fate partire il programma, e vedrete i vari byte riempirsi secondo la loro sequenza in memoria.

Lateralmente...

Per spostare lateralmente tutta l'immagine video potremo perciò procedere sequenzialmente, ossia spostando un byte dopo l'altro così come li troviamo, senza andarli a pescare qua e là. Certo, non sposteremo l'immagine seguendo un ordinato schema dall'alto verso il basso, ma alla fine l'avremo spostata tutta, ed è questo ciò che importa.

Il listato 2 esegue uno scrolling verso sinistra di un carattere. Per questo esempio, come per il precedente, è stato usato il BASIC, in modo da ridurre decisamente la velocità. Il meccanismo di azione risulta così chiaramente visibile. In questo caso potete vedere l'immagine che dapprima si scompone, mano a mano che le varie linee vengono spostate. Solo al termine dell'operazione vedremo l'immagine ricomposta, spostata a sinistra rispetto all'originale di un carattere (cioè di un byte).

Quello dell'esempio è uno spostamento a passi di un carattere. Vediamo di realizzare la stessa cosa in linguaggio macchina. La versione BASIC è infatti, come abbiamo constatato, inaccettabilmente lenta. Una versione in l/m, viceversa, sarebbe così veloce da non rendere evidente il procedimento di "demolizione e ricostruzione" dell'immagine.

La prima routine

Nel listato 3 troviamo appunto una routine, scritta in Assembly, per lo spostamento a sinistra dell'immagine a passi di un byte. Vediamo di analizzarne il funzionamento.

Una descrizione sommaria del suo meccanismo potrebbe essere questa:

— per ciascuna riga:

-- carica il contenuto del secondo elemento nel primo, il contenuto del terzo nel secondo e così via, fino a caricare il contenuto dell'ultimo elemento della riga nel penultimo.

-- Carica zero nell'ultimo elemento della riga.

```
.  
.  
.*  
*listato ASSEMBLY  
*della routine  
*di spostamento  
*laterale a sinistra  
*a passi di un byte  
*  
org 50001  
.  
.  
inizializzazioni  
.  
50001 21 01 40      ld hl,16385  
50004 11 00 40      ld de,16384  
50007 3E C0          ld a,192  
.  
.  
ciclo  
per ciascuna linea  
.  
LOOP  
50009 01 1F 00      ld bc,31  
50012 ED B0          ld ir  
.  
.  
azzeramento ultimo  
byte di ogni linea  
.  
50014 EB            ex de,hl  
50015 36 00          ld (hl),0  
50017 EB            ex de,hl  
.  
.  
decremento  
contatore  
e controllo  
fine ciclo  
.  
50018 3D            dec a  
50019 C8            ret z  
.  
.  
incremento  
puntatori  
e chiusura ciclo  
.  
50020 13            inc de  
50021 23            inc hl  
50022 18 F1          jr LOOP  
.  
.  
.  
.  
.  
.  
.  
.  
33,  1, 64, 17,  0, 64, 62, 192,  
1, 31,  0, 237, 176, 235, 54,  0,  
235, 61, 200, 19, 35, 24, 241
```

```

*****
listato ASSEMBLY
della routine
di spostamento
laterale a destra
a passi di un byte
*****
org 50001
-----
inizializzazioni
-----
50001 21 FE 57      ld hl,22526
50004 11 FF 57      ld de,22527
50007 3E C0         ld a,192
-----
ciclo
per ciascuna linea
-----
LOOP
50009 01 1F 00      ld bc,31
50012 ED B8         lddr
-----
azzeramento ultimo
byte di ogni linea
-----
50014 EB           ex de,hl
50015 36 00        ld (hl),0
50017 EB           ex de,hl
-----
decremento
contatore
e controllo
fine ciclo
-----
50018 3D           dec a
50019 C8           ret z
-----
decremento
puntatori
e chiusura ciclo
-----
50020 1B           dec de
50021 2B           dec hl
50022 18 F1        jr LOOP
-----
-----
33,254, 87, 17,255, 87, 62,192,
1, 31, 0,237,184,235, 54, 0,
235, 61,200, 27, 43, 24,241

```

- La riga era l'ultima?
SI =fine NO=proseguì con la riga seguente

Per compiere questo "riversamento di contenuti" da un byte al precedente, per tutta una riga, useremo un'operazione nuova, molto potente:

ldir

Il suo significato è "load, increment, return if zero", cioè carica, incrementa e ritorna se zero.

In parole povere, questa operazione prende il contenuto dell'indirizzo indicato da hl e lo copia nell'indirizzo indicato da de. Quindi incrementa sia hl che de, mentre decrementa invece bc. A questo punto ripete la stessa operazione sui nuovi indirizzi, che saranno ambedue superiori di uno rispetto ai precedenti. Si va avanti così finché bc, a furia di essere decrementata, non giunge a zero. A questo punto l'operazione è conclusa, e l'esecuzione passa all'operazione seguente.

Una singola operazione

ldir

equivale in pratica a

```

CICLO  ld a,(hl)
        ld (de),a
        inc hl
        inc de
        dec bc
        jr nz, CICLO

```

Oltretutto, per un difetto dello Z80, l'operazione

dec bc

non influenza i flag, per cui sarebbe necessario aggiungere, per rendere funzionante il blocco:

```

ld a,b
or c

```

fra la penultima operazione (dec bc) e l'ultima (jr nz, CICLO), allo scopo di riflettere sul flag di zero il contenuto di bc.

Esiste, nel vasto repertorio di operazioni dello Z80, anche l'operazione singola

ldi

che esegue una sola volta il trasferimento dall'indirizzo puntato da hl a quello puntato da de, e quindi il loro incremento. Questo può essere utile volendo ad esempio usare un contatore diverso da bc, magari perché, essendo il numero di byte da modificare minore di 256, un singolo registro è sufficiente.

In questo caso si potrebbe usare il blocco:


```
ret z
inc de
inc hl
jr LOOP
```

Notiamo subito che questa routine è identica a quella per spostare a sinistra l'immagine. Cambia solo l'indirizzo di partenza e il numero delle righe (prima in realtà erano linee). La cosa più importante è che la lunghezza di ciascuna riga è uguale a quella delle linee dell'area immagine. Perché, allora, non unire le due cose, realizzando una routine che sposti un blocco di memoria comprendente le due aree?

Tutto insieme

Il risultato di questa operazione lo vedete nei due listati che riportano l'Assembly delle routine per il movimento simultaneo (non è simultaneo, ma sequenziale, in realtà; la velocità è però talmente alta da dare questa impressione) di immagine ed attributi.

Non è il caso di esaminarne di nuovo il funzionamento, ormai a noi ben noto. Basta osservare la differenza nei valori di inizializzazione, che qui sono in cima all'area immagine per lo spostamento a sinistra, ed in fondo all'area attributi per quello a destra. Il contatore di linea (o riga, a seconda del caso), a, parte adesso da

```
216=192+24
```

cioè la somma delle linee di immagine e delle righe di attributi.

Vogliamo provare ad usare le due nuove routine?

Dopo averne caricato il codice macchina (lo trovate nelle figure) creiamo uno schermo pieno di figure e colori.

Supponiamo di aver caricato all'indirizzo 40000 la routine di scrolling a sinistra, ed all'indirizzo 50000 quella di scrolling a destra; vogliamo spostare l'immagine a sinistra di cinque caratteri?

```
FOR i=1 TO 5
RANDOMIZE USR 40000
NEXT i
```

Vogliamo spostarla a destra di otto?

```
FOR i=1 TO 8
RANDOMIZE USR 50000
NEXT i
```

Facciamo ora un balzo in avanti e prefiggiamoci uno scopo ben più ambizioso: lo spostamento laterale in alta risoluzione, ovvero pixel per pixel.

Detto così potrebbe anche sembrare qualcosa di complesso, ma vedremo che non è così.

Pixel per pixel

Il problema è simile al precedente, ma a un livello diverso; si tratta di far scorrere tutti i byte di una linea (una delle 192 dell'area immagine) a destra o a

sinistra di una posizione. Questo comporta una specie di "passamano" fra i vari byte. Nel caso dello scrolling a sinistra, ad esempio, il primo byte perderà il suo bit più a sinistra, cioè il 7, e ne prenderà uno dal byte alla sua destra. Ovviamente, il bit scambiato sarà quello più a sinistra cioè il 7. Proseguendo di questo passo si giunge all'ultimo byte della linea, che cederà il bit 7 al suo vicino di sinistra e subirà l'azzeramento del bit 0.

In pratica, tutto questo avviene per mezzo delle operazioni di scorrimento di registro, già viste la scorsa volta. Si tratta di

```
sl <registro>
```

```
ed
```

```
sr <registro>
```

che fanno scorrere i bit all'interno del registro indicato, rispettivamente a sinistra o a destra. In entrambi i casi il bit che viene scartato (il 7 nel caso di sl, lo 0 nel caso di sr) finisce nel flag di riporto, che cede il suo precedente contenuto al bit opposto, che deve essere riempito.

Per mezzo della ripetizione di una di queste due operazioni è perciò possibile realizzare una catena di scorrimento, in cui il primo byte a subire lo scorrimento cede un suo bit al flag di riporto, che a sua volta lo cede al byte successivo, che scarterà un bit ponendolo a sua volta nel flag, per essere quindi recuperato dal byte ancora successivo e così via.

Scorrimento a sinistra

Vediamo la prima routine, quella di scorrimento a sinistra.

Dovendo far scivolare a sinistra il contenuto dei byte che compongono ciascuna linea, è necessario iniziare da destra. Questo perché ciascun byte, scorrendo, possa trovare nel flag di riporto il bit scartato dal byte alla sua destra. In caso contrario, cioè partendo da sinistra, ad essere recuperato sarebbe il bit scartato dal byte a sinistra, e questo produrrebbe un colossale miscuglio.

Noi useremo hl come puntatore che si sposta all'interno dell'area immagine, e di conseguenza le operazioni

```
rl (hl)
```

```
ed
```

```
rr (hl)
```

Il puntatore viene innanzitutto inizializzato a 22527, ultimo byte dell'area, mentre il contatore di linea, c in questo caso, viene posto come sempre a 192.

A questo punto per ciascuna linea si ripete il medesimo ciclo, composto dalla sequenza iniziale

```
ld b, 32
and a
```

che inizializza il contatore di colonna ed azzerà il flag di riporto. Per questo azzeramento viene utilizzata l'operazione

```
and a
```

Tutte le operazioni di and hanno infatti l'effetto secondario di azzerare il flag di riporto. Nel nostro caso il registro a non è utilizzato dalla routine, ma in ogni caso l'operazione di and con se stesso non danneggerebbe il suo contenuto. L'unico effetto di

```
and a
```

è infatti quello di azzerare il flag di riporto.

Segue il ciclo di colonna, che si ripete per 32 volte ad ogni linea.

È composto dall'operazione di scorrimento, in questo caso

```
rl (hl)
```

seguita dall'aggiornamento del puntatore

```
dec hl
```

e dalla chiusura del ciclo.

Avendo azzerato il flag di riporto prima del primo scorrimento, il byte più a destra di ciascuna linea caricherà il suo bit 0 con zero e cederà il suo bit 7 al flag. La catena di recuperi e cessioni, che abbiamo già esaminato, proseguirà fino al byte più a sinistra, che lascerà nel flag il suo bit 7.

L'eventuale successivo ciclo di colonna, comunque, si sbarazzerà di questo bit con l'operazione iniziale

```
and a
```

Al termine di ciascun ciclo di linea troviamo l'aggiornamento del contatore c, che viene decrementato, ed il controllo sull'eventuale esaurimento delle linee, con

```
ret z
```

In caso contrario il ciclo si chiude con

```
jr LOOP2
```

Nel listato i due cicli sono evidenziati da frecce, che ne marcano gli estremi.

Scorrimento a destra

Per lo scorrimento a destra il discorso è lo stesso, capovolto.

Si parte dall'estremo sinistro di ciascuna linea, e quindi dal primo byte dell'area. Il puntatore hl viene perciò inizializzato a 16384.

Uguale è il valore attribuito a c, e così pure a b. All'inizio del ciclo di linea l'operazione

```
and a
```

azzerà il flag di riporto.

All'interno del ciclo di colonna troviamo le uniche differenze:

rr (hl)
inc hl

Qui lo scorrimento è verso destra, ed infatti è il bit 7 del byte più a sinistra a raccogliere il valore zero creato nel flag dall'operazione and a.

Dovendo procedere da sinistra a destra, inoltre, il puntatore hl viene aggiornato con un incremento.

Il bit che viene scartato è questa volta lo 0 del byte più a destra di ciascuna linea.

La sezione finale della routine è identica alla precedente.

In ambedue i casi trovate nelle figure i codici macchina in forma decimale, pronti per essere introdotti.

Anche qui ci troviamo in presenza di routine rilocabili, non essendo stato fatto uso di salti assoluti né di subroutine.

Ciascuna chiamata, effettuata mediante

RANDOMIZE USR <indirizzo>

ha l'effetto di spostare l'intera immagine video a sinistra o a destra di un pixel. Volendo perciò muoverla di un numero n di pixel, occorrerà ricorrere, come nel caso precedente, a un ciclo FOR...NEXT.

Niente attributi!

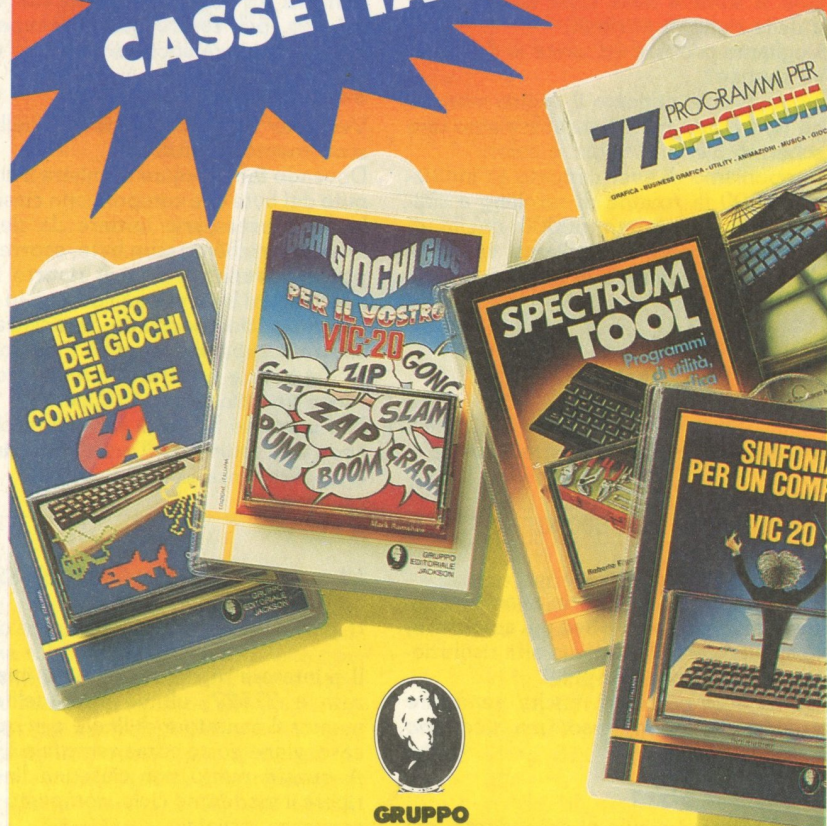
In questo caso è esclusa la possibilità di spostare l'area attributi. È noto infatti che nello Spectrum tutti i pixel di uno stesso carattere dividono gli stessi attributi, ed è perciò impossibile mantenere la coerenza colore-immagine durante lo scrolling ad alta risoluzione. Normalmente si ovvia a questo inconveniente tenendo lontani i vari caratteri e adottando uno sfondo uniforme per tutta l'immagine. In questo modo è possibile, ogni otto spostamenti di un pixel, chiamare una volta una routine che muova nel senso appropriato gli attributi, senza che nel frattempo si sia notato uno squilibrio. Lascio a voi l'onere di stendere una routine di questo tipo, ricordando che ne abbiamo parlato in abbondanza più su, arrivando a "quasi scriverla".

Per questa volta ci fermiamo qui. La prossima volta vedremo, unitamente ad altre interessanti routine, un modo per spostare l'immagine a passi di mezzo byte.

A presto!

Novità firmate Jackson.

CON CASSETTA



GRUPPO
EDITORIALE
JACKSON

La biblioteca che fa testo.

REM:HW

Hardware

Un tasto dolente? Sostituiamolo

di **Marcello Spero**

Dopo il discorso introduttivo della volta scorsa sulle prospettive legate a una "professionalizzazione" più o meno accentuata del nostro Spectrum, ed una presentazione delle problematiche legate a quello che ci sembra sia il primo e più necessario passo da compiere, e cioè l'affiancamento di una tastiera "vera" a quella originale, iniziamo in questa puntata la parte pratica.

Ricapitoliamo rapidamente i nostri obiettivi.

Abbiamo parlato di affiancamento, e non di sostituzione, di una nuova tastiera a quella originale. Questo significa che non intendiamo in alcun modo disinserrire, coprire o comunque disattivare la "vecchia" tastiera.

Il connettore interno, utilizzato da molte case produttrici di tastiere sostitutive per lo Spectrum per collegare i loro prodotti senza bisogno di alcun dispositivo supplementare, non può essere usato. Questo per due ragioni:

- è già impegnato con i collegamenti alla tastiera originale, e non potrebbe sopportare ulteriori connessioni;
- un suo collegamento con un'unità esterna allo Spectrum, qual è appunto una tastiera, comporterebbe la necessità di forare o mantenere aperto il contenitore della macchina, cosa che vogliamo assolutamente evitare.

Non ci rimane quindi che ricorrere a un nuovo decodificatore, che ci consenta di collegare la nostra tastiera direttamente al connettore posteriore dello Spectrum.

Come abbiamo visto la scorsa volta, i segnali da intercettare sono parecchi.

Possiamo dividerli in due categorie:

- segnali comuni a tutte le richieste di lettura della tastiera;
- segnali che indicano quale sezione della tastiera deve essere letta.

Alla prima categoria appartengono quei segnali che devono essere tutti contemporaneamente presenti per indicare la volontà da parte della CPU di ricevere i segnali della tastiera.

Sono:

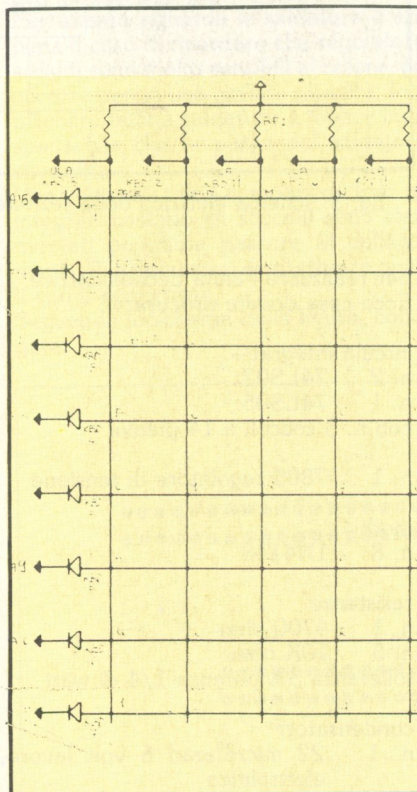


fig. 1: schema della tastiera dello Spectrum

IORQGE

RD

A0

e per indicare una richiesta di lettura di tastiera devono essere tutti e tre a livello basso.

Alla seconda appartengono invece le linee indirizzi

A8 - A15

Ad ognuna di esse corrisponde un gruppo di cinque tasti, detto semiriga, in questo modo:

linea	semiriga
A9	SPACE-B
A10	ENTER-H
A11	P-Y
A12	0-6
A13	1-5
A14	Q-T
A15	A-G
A16	SHIFT-V

Anche qui la selezione della semiriga avviene con la presenza di un livello logico basso sulla relativa linea indirizzi.

In risposta alla richiesta di lettura di una semiriga la tastiera invierà sul bus dati un byte, i cui bit da 0 a 4 indicheranno lo stato dei cinque tasti della semiriga indicata. Per ciascuna semiriga D0 indicherà lo stato del tasto più esterno, D4 quello del tasto più interno. A tasto premuto corrisponderà un livello basso, altrimenti il livello sarà alto.

Tutto questo avviene in modo estremamente semplice ed economico, nel tipico stile Sinclair. La tastiera è organizzata come una matrice di linee dati e indirizzi (lo vedete in figura), ed i vari tasti altro non fanno che mettere in contatto la linea indirizzi relativa alla loro semiriga con la linea dati relativa alla

loro colonna. Le linee dati sono tenute a livello alto da apposite resistenze, tali da consentire l'instaurarsi di un livello logico basso solo in caso di collegamento con una linea indirizzi bassa, cioè selezionata.

Quindi...

Il lavoro di decodifica si riduce a collegare le linee dati provenienti dalla tastiera con quelle della CPU solo in presenza di un livello basso sulla terna IORQGE, RD e A0.

Questo compito viene svolto normalmente dall'ULA, per la tastiera originale.

Noi, volendo collegare una nuova tastiera senza sfruttare il connettore interno (pilotato appunto dall'apposito circuito integrato), dobbiamo realizzare un circuito esterno di decodifica.

Lo schema di un circuito adatto a questo scopo è riportato in figura, ed è già stato discusso la scorsa volta. Si tratta in sostanza di una combinazione di porte logiche NOR e NOT, collegate in modo da produrre, al termine della catena, un livello logico basso solo se all'inizio della catena stessa tutti i segnali sono a livello basso.

Ricordando che la tabella di verità di una porta NOR è

ingresso 1	ingresso 2	uscita
basso	basso	alta
basso	alto	bassa
alto	basso	bassa
alto	alto	bassa

mentre una porta NOT (detta anche invertitore) si limita a invertire il segnale in ingresso (da alto a basso e viceversa), seguiamo brevemente la catena, con un occhio allo schema.

IORQGE e A0 a livello basso provocano in uscita di IC1a (NOR) un livello alto, che viene trasformato in basso da IC3a (NOT). L'uscita di IC3a va a costituire uno dei due ingressi di IC1c (NOR); l'altro ingresso è RD. Se ambedue sono a livello basso, all'uscita di IC1c troveremo un livello alto, trasformato in basso da IC1d. Questa porta sarebbe in realtà un NOR, ma viene trasformata in NOT con il collegamento in parallelo dei suoi ingressi, per utilizzare tutte le porte presenti nei "chip" che utilizziamo.

In uscita dalla catena avremo quindi un livello basso, che viene "iniettato" in ingresso alle cinque porte che controllano il passaggio dei segnali lungo le linee dati. Se questo ingresso è alto, l'uscita resterà bassa, senza alcun riguardo alle variazioni di livello sull'altro ingresso. Questo livello alto viene trasformato in basso dalle porte NOT. Solo in presenza di un livello basso le uscite seguiranno le variazioni dell'altro ingresso.

La lista della spesa

Vediamo ora in pratica cosa ci occorre

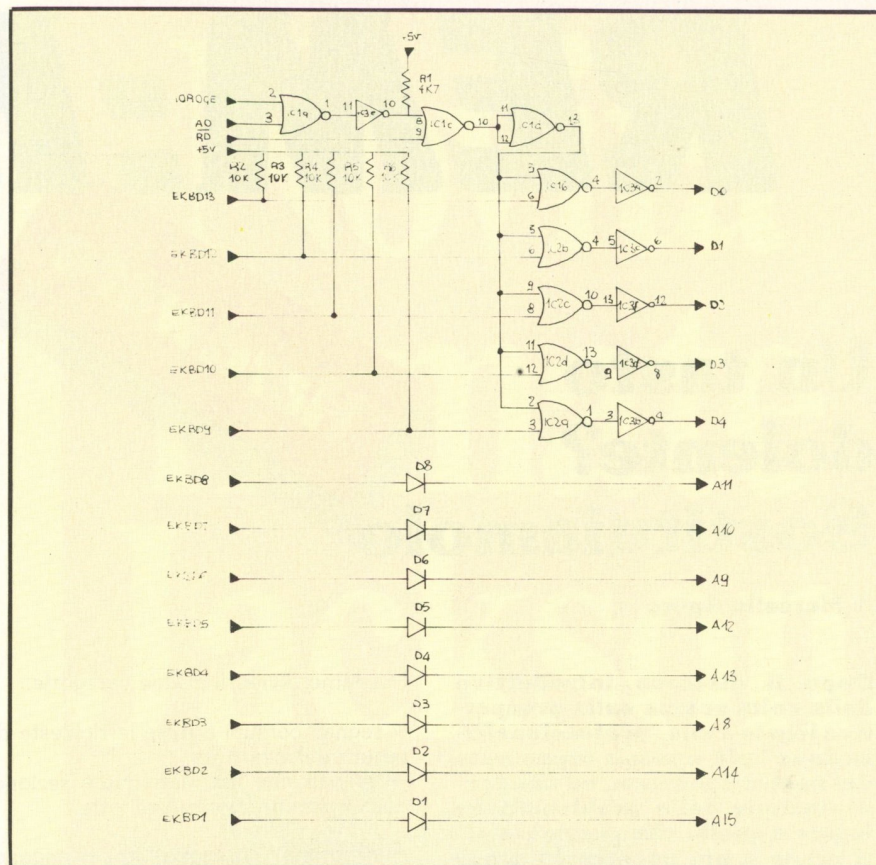


fig. 2: schema elettrico del decodificatore

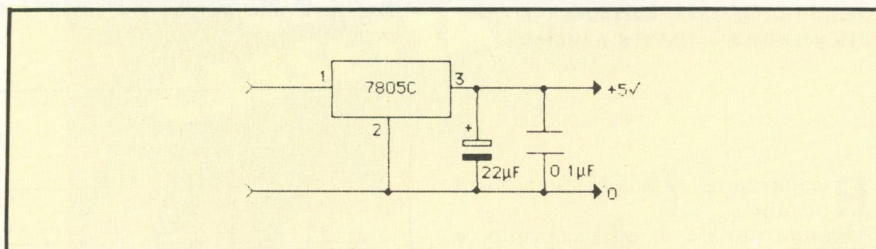


fig. 3: schema elettrico della sezione alimentatrice

per realizzare l'unità decodificatrice. Ecco cosa dovete procurarvi:

- circuiti integrati
- n. 2 74LS02
- n. 1 74LS05
- con n. 3 zoccoli a 14 piedini
- n. 1 7805 regolatore di tensione
- diodi
- n. 8 1N4148
- resistenze
- n. 1 4700 ohm
- n. 5 10K ohm
- tolleranza 5% potenza 1/4 di watt

- condensatori
- n. 1 22 microfarad 6 volt lavoro, elettrolitico
- n. 1 0.1 microfarad ceramico a disco
- n. 1 connettore a 28 vie passo 2.54

mm (vedi testo)

n. 1 basetta per montaggi sperimentali, "a bollini", di circa 10x5 cm

un saldatore, con relativo stagno

qualche metro di filo elettrico isolato, il più sottile possibile

circa 50 cm di filo elettrico isolato di sezione un po' più grossa

circa un metro di cavo piatto a 15 poli (oppure due spezzoni da un metro di cavo piatto a otto poli, come vedete nelle foto).

Il montaggio

La tecnica di montaggio più comoda, usando una basetta a bollini, è quella di saldare in posizione i componenti, e quindi collegarli fra loro con il filo.

Il primo componente da collocare al suo posto è il connettore a 28 poli, quello

che consentirà il collegamento con lo Spectrum.

In Italia non è facile reperire un connettore passo 2.54 mm con le estremità aperte. Molto più semplice è invece l'acquisto di un connettore di passo corretto, ma con un numero superiore di contatti e con le estremità chiuse.

La cosa da fare a questo punto è segare via le estremità ed i contatti in più, servendosi di un comunissimo seghetto manuale. È buona norma, prima di iniziare il lavoro, estrarre i contatti delle posizioni in cui si lavorerà con la sega. Il metallo potrebbe infatti danneggiare il filo. Una volta ottenuto un connettore delle giuste dimensioni e ad estremità aperte estraete il contatto metallico dalla quinta posizione, a partire da una delle due estremità, per creare la fessura di centraggio.

Ora dovrete procurarvi un pezzetto di qualcosa di corrette dimensioni e spessore, da inserire nella fessura. Lo spessore adatto ce l'ha una qualsiasi basetta per circuiti stampati; il problema è quello di tagliarne un pezzetto delle dimensioni giuste. Anche un pezzetto di plastica o cartone molto rigido può andare bene, purché lo spessore sia sufficiente e non consenta gioco.

È giunto il momento di saldare in posizione il connettore. Il posto ideale è a filo di uno dei lati lunghi della basetta. Anche se non verranno utilizzati tutti i contatti, è bene che le saldature ci siano tutte, per ragioni di solidità.

Dopo aver sistemato il connettore è la volta degli zoccoli per gli integrati.

Potete metterli dove volete, ma tenete conto del fatto che per inserire il connettore in modo appropriato allo Spectrum una fascia di circa 2.2 cm al di sopra di esso deve essere libera da sporgenze pronunciate. In pratica, in questa zona ci possono essere solo fili, ma non componenti. Gli zoccoli andranno quindi collocati nella parte alta della basetta, magari tutti nello stesso senso e con la posizione del piedino 1, che è sempre marcata, dallo stesso lato, per facilitare le fasi successive del montaggio.

Sarà quindi la volta dei diodi, da mettere in fila e con la fascia scura dallo stesso lato, e delle resistenze, le cinque di valore uguale tutte in fila, quella diversa un po' staccata.

Il capitolo alimentazione merita un discorso a parte. I componenti destinati a questo scopo sono i due condensatori ed il regolatore 7805. Nel prototipo (che essendo tale non è purtroppo un modello di ordine e razionalità) che vedete nelle foto essi sono stati collocati dalla parte del rame, non essendo rimasto posto che nella zona "proibita" dove non devono esserci sporgenze (dimenticavo: se possedete l'interfaccia 1, non ci sono "zone proibite" da lasciare libere). Non mi sembra un'idea eccellente; disponendo di una basetta di adeguate dimensioni è il caso di montarli dalla parte giusta.

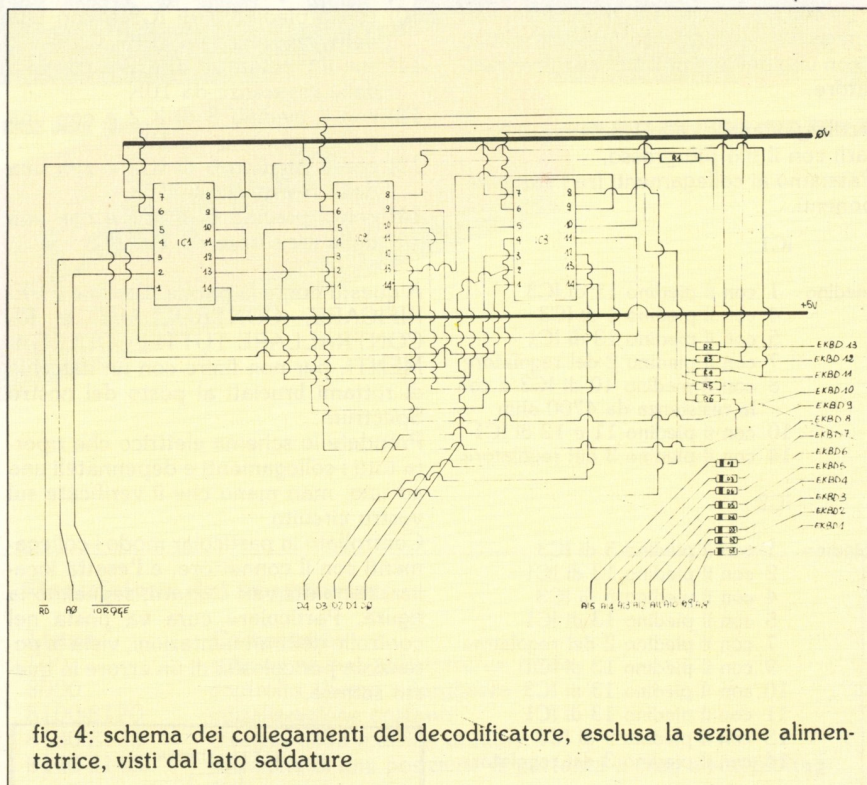


fig. 4: schema dei collegamenti del decodificatore, esclusa la sezione alimetrica, visti dal lato saldature

In previsione dei collegamenti saldate i tre componenti in fila, in modo che sia agevole unire il piedino del regolatore con il positivo del condensatore elettrolitico (è marcato) ed uno dei due poli dell'altro condensatore, ed il piedino 2 con gli altri due poli dei condensatori. Per quanto riguarda le saldature è appena il caso di ricordare che regolatore e diodi sono molto sensibili al calore, ed occorre perciò una grande rapidità. Per ottenere buone saldature è inoltre indispensabile che la punta del saldatore sia pulita.

Una volta che tutti i componenti (per gli integrati bastano gli zoccoli) sono stati montati possiamo passare ai collegamenti. Per primi direi di effettuare quelli con il connettore.

Seguendo lo schema della figura, collegate:

- A8-A15 con il polo negativo (lato con la fascia) di D1-D8
- A0 con il piedino 3 di IC1
- IORQGE con il piedino 2 di IC1
- RD con il piedino 9 di IC1
- D0 con il piedino 2 di IC3
- D1 con il piedino 6 di IC3
- D2 con il piedino 12 di IC3
- D3 con il piedino 8 di IC3
- D4 con il piedino 4 di IC3
- +9V con il piedino 1 del regolatore 7805

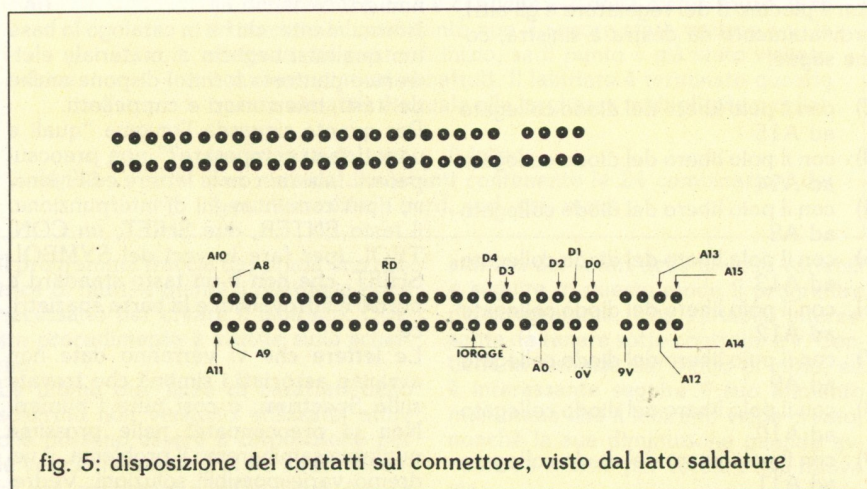


fig. 5: disposizione dei contatti sul connettore, visto dal lato saldature

– 0v con il piedino 2 del regolatore, con il negativo del condensatore elettrolitico e con uno dei due poli dell'altro condensatore.

I collegamenti di alimentazione è bene farli con il filo più grosso. Passiamo ai collegamenti fra i vari componenti.

IC1

- piedino 1 con il piedino 11 di IC3
- 4 con il piedino 1 di IC3
- 5 con il piedino 13 di IC1
- 7 con il piedino 2 del regolatore
- 8 con il piedino 10 di IC3 e con la resistenza da 4700 ohm
- 10 con il piedino 11 e 12 di IC1
- 14 con il piedino 3 del regolatore

IC2

- piedino 1 con il piedino 3 di IC3
- 2 con il piedino 13 di IC1
- 4 con il piedino 5 di IC3
- 5 con il piedino 13 di IC1
- 7 con il piedino 2 del regolatore
- 9 con il piedino 13 di IC1
- 10 con il piedino 13 di IC3
- 11 con il piedino 13 di IC1
- 13 con il piedino 9 di IC3
- 14 con il piedino 3 del regolatore

IC3

- piedino 7 con il piedino 2 del regolatore
- 14 con il piedino 3 del regolatore

7805 (regolatore)

- piedino 3 con il polo positivo del condensatore elettrolitico e con il polo ancora libero dell'altro condensatore

- 10) con il piedino 3 di IC2 e con una delle resistenze da 10K
- 11) con il piedino 12 di IC2 e con una delle resistenze da 10K
- 12) con il piedino 8 di IC2 e con una delle resistenze da 10K
- 13) con il piedino 6 di IC2 e con una delle resistenze da 10K
- 14) con il piedino 6 di IC1 e con una delle resistenze da 10K

A questo punto la parola d'ordine è VERIFICARE, CONTROLLARE e RICONTROLLARE TUTTI I COLLEGAMENTI, per non finire con un mucchio di rottami bruciati al posto del nostro Spectrum.

Prendete lo schema elettrico che riporta tutti i collegamenti e depennateli uno ad uno, man mano che li verificate sul vostro circuito.

Controllate in particolar modo i collegamenti con il connettore, e l'esatta localizzazione dei vari contatti, seguendo la figura. Particolare cura va posta nel controllo delle alimentazioni, vista la potenziale pericolosità di un errore in questo settore.

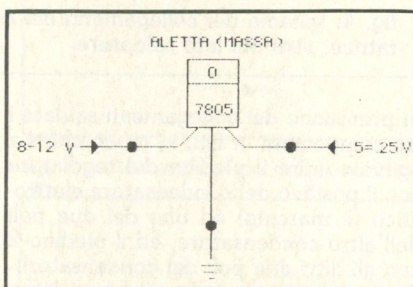


fig. 6: il regolatore 7805

Tanti tasti tosti

Per impegnare la vostra mente con qualcosa di diverso, mentre procede la verifica (meglio, le verifiche) del decodificatore, iniziate ad interessarvi per l'acquisto dei componenti meccanici della tastiera.

Ciò che occorre, lo vedete nelle foto, è una base in metallo opportunamente forata, un certo numero di interruttori ed i relativi cappellotti con le lettere ed i numeri.

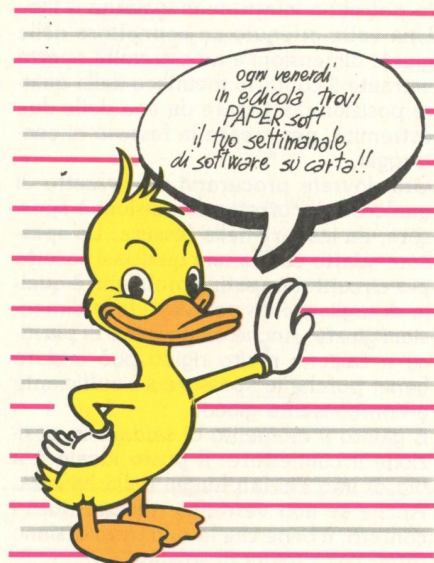
Normalmente, chi ha in catalogo la base (un qualsiasi negozio di materiale elettronico piuttosto fornito) dispone anche dei tasti, interruttori e cappellotti.

Per quanto riguarda l'enigma "quali e quanti tasti comperare?" non preoccupatevi. Iniziate con le lettere ed i numeri, i più comuni segni di interpunzione, il tasto ENTER, due SHIFT, un CONTROL (per fare le veci del SYMBOL SHIFT, che non è un tasto standard e perciò è introvabile) e la barra spaziatrice.

Le lettere che vi verranno date non avranno associati i simboli che trovate sullo Spectrum, e così pure i numeri. Non vi preoccupate: nelle prossime puntate esamineremo il problema, e vedremo varie possibili soluzioni. Vedre-

mo anche come sia possibile avere tasti che esistono solo sullo Spectrum Plus (cursori, extended mode, delete), realizzando il doppio contatto per via elettronica.

Per questa volta, comunque, basta così; arrivederci!



Ci siamo quasi

L'ultima fase dei collegamenti riguarda le uscite verso la tastiera. Che usiate un tratto unico di cavo a 15 poli o due spezzoni da otto (in realtà sarebbero sufficienti uno da otto e uno da sette, ma non ci risulta reperibile cavo piatto a sette poli) la cosa non cambia. Salderemo il conduttore più a destra al piedino 2 del regolatore, quello più a sinistra con il piedino 3 del regolatore e gli altri, ordinatamente da destra a sinistra, come segue:

- 2) con il polo libero del diodo collegato ad A15
- 3) con il polo libero del diodo collegato ad A14
- 4) con il polo libero del diodo collegato ad A8
- 5) con il polo libero del diodo collegato ad A13
- 6) con il polo libero del diodo collegato ad A12
- 7) con il polo libero del diodo collegato ad A9
- 8) con il polo libero del diodo collegato ad A10
- 9) con il polo libero del diodo collegato ad A11

La cassetta allegata a questo numero di Supersinc contiene in omaggio un programma realizzato per lo Spectrum "Superscrol" del Gruppo Utilizzatori Computer Sinclair Napoli c/o Roberto Chimenti via Luigi Rizzo 18 - 80124 Napoli (NA) - Tel. 081/7623121.

Labirinti

di **Danny O'Mara**
trad. e adatt.
di **Marcello Spero**

Vi siete mai chiesti come far costruire un labirinto dal computer?

Questo breve programma traccia sullo schermo labirinti casuali, ed è stato scritto per dimostrare il procedimento di creazione di un labirinto.

Sebbene il programma in sé non sia lungo, la memoria richiesta per i suoi dati è notevole, e per un corretto funzionamento è quindi richiesto uno Spectrum 48K.

L'altezza e la larghezza del labirinto possono comunque essere ridotte, nel qual caso il tutto sarà utilizzabile anche dalla versione 16K. Questo può essere ottenuto modificando le variabili *w* e *h* in linea 10. Per i migliori risultati la larghezza *w* deve essere compresa tra 15 e 31 e l'altezza *h* tra 10 e 19.

Dopo il caricamento il programma "medita" per circa due minuti, e quindi traccia sullo schermo una serie di labirinti dal percorso casuale, copiandoli su stampante se questa è collegata. Le pareti del labirinto sono disegnate in nero, mentre i percorsi sono bianchi. Le posizioni di inizio e di fine sono marcate con un puntino.

I labirinti realizzati da questo programma sono detti "a singola connessione". In un labirinto di questo tipo esiste un'unica strada possibile fra due punti qualsiasi, che è come dire che non ci sono anelli. Si tratta del tipo più semplice di labirinto generabile da computer. Il labirinto a percorso multiplo richiede programmi di generazione molto più complessi.

Tanto per citare un esempio, i giochi tipo Pacman utilizzano semplici labirinti a connessione multipla. Ovviamente, un labirinto a singola connessione non può essere usato per un Pacman: non ci sarebbe via di scampo!

Lo schermo è suddiviso in blocchi di 4*4 pixel. Un labirinto costruito usando semplicemente le 32*22 posizioni dei caratteri sarebbe infatti troppo facile da risolvere.

Descrizione del programma

Linee	Commento
100	subroutine per abbattere un muro
8000	controllo altezza e larghezza
8100-8120	inizializzazione costanti
8130-8135	disegno della matrice di muri
8140-8145	selezione di una posizione di partenza a caso e suo plottaggio
8150	inizio del ciclo principale
8160	selezione di una delle 24 possibili serie di direzioni
8165-8172	prova della prima direzione e controllo per verificare se conduce allo stesso punto di prima. Se questo succede, il programma salta alla linea 8200, dove viene selezionata una nuova direzione
8175-8178	il punto di arrivo è nuovo; abbattimento di un muro e controllo per l'eventuale completamento del labirinto
8880-8890	memorizzazione della massima distanza dal punto di partenza e ripetizione del ciclo principale
8200	selezione di un'altra direzione da provare, se non sono già state tentate tutte e quattro
8210-8222	se tutte le direzioni sono state provate, viene ripercorso all'indietro il labirinto finché è possibile una nuova direzione
8300	fine; plottaggio della posizione più distante dal punto di inizio

Descrizioni delle variabili

<i>w</i>	larghezza del labirinto; fino a un massimo di 31
<i>h</i>	altezza del labirinto; fino a un massimo di 19
<i>r,s</i>	larghezza più altezza in termini di blocchi 4*4 pixel
<i>a()</i>	ha dimensioni <i>r</i> ed <i>s</i> e contiene -1 dove c'è un muro, 0 se il tracciato in quel punto non è ancora stato visitato o <i>v</i> , la distanza dal punto di inizio, se il punto è già stato visitato
<i>done</i>	il numero di muri abbattuti. Il labirinto è terminato quando questa variabile è uguale alla larghezza moltiplicata per l'altezza
<i>v</i>	la distanza dal punto di inizio
<i>d(4,24)</i>	una matrice di costanti contenente le 24 combinazioni dei 4 punti cardinali: nord, sud, est, ovest

Il programma traccia dapprima una matrice di muri, e quindi li distrugge dove necessario per creare il labirinto. Questo procedimento è visibile sullo schermo.

Le ultime due linee di caratteri dello schermo non vengono utilizzate, affinché possano essere a disposizione per le vostre necessità. Potete infatti inserire nel programma delle linee per la

stampa dei valori delle diverse variabili e seguire in questo modo il progredire del programma. Una variabile interessante da tenere sotto controllo è *v*. Contiene la distanza dal punto di inizio, ed è interessante seguire il suo aumento man mano che il labirinto viene creato, nonché la sua diminuzione quando avviene un ritorno per cercare una nuova via.

Qui di seguito viene fornita una descrizione più dettagliata del programma, per permettere la comprensione del suo funzionamento. Prima di tutto, però, ecco alcuni suggerimenti per eventuali modifiche.

- Il programma può di tanto in tanto produrre lunghi tracciati con brevi diramazioni cieche, rendendo troppo facile la soluzione. Per evitare il prodursi di situazioni di questo tipo, la variabile v può essere utilizzata per limitare la lunghezza di singoli tratti di percorso. Que-

sta variabile può essere confrontata con un valore casuale, ed in caso di superamento di quest'ultimo può essere forzato il ritorno alla ricerca di una nuova via.

- Una modifica piuttosto semplice è l'aggiunta sullo schermo di un cursore controllato dai tasti 5-8 dello Spectrum. Questi tasti possono essere usati per muovere il cursore all'interno del labirinto e cercare di risolverlo.

- Nello Spectrum 48K la memoria è sufficiente a creare labirinti composti da

blocchi di 2*2 pixel. Questo, insieme a una lunghezza casuale dei vari rami, potrebbe dare origine a labirinti molto complessi.

- Per i più esperti, cosa ne dite di un programma che riesca a risolvere il labirinto?

Le altre variabili usate in questo programma servono per la conservazione temporanea di risultati intermedi, al fine di aumentare la velocità del programma.

Labirinti

versione per ZX Spectrum 48K

```

1 REM Labirinto
10 LET w=31: LET h=19: GO TO 8000
100 LET px=xx*4-4: LET py=yy*4+12
102 OVER 1: FOR z=0 TO 3: PLOT px+z,
py: DRAW 0,3: NEXT z: OVER 0
104 RETURN
8000 LET r=2*w+1: LET s=2*h+1: IF r>6
3 OR s>39 OR r<10 OR s<10 THEN
PRINT AT 0,0;"Height or width o
ut of range.": GO TO 9999
8100 DIM a(r,s)
8105 DIM d(4,24): RESTORE 9000: FOR i
=1 TO 24: FOR j=1 TO 4: READ d(j
,i): NEXT j: NEXT i
8110 DIM i(4): DIM j(4): DIM k(4):
DIM l(4): FOR i=1 TO 4: READ i(
i): READ j(i): READ k(i): READ l
(i): NEXT i
8115 FOR i=1 TO r: FOR j=1 TO s:
LET a(i,j)=-1: NEXT j: NEXT i
8120 FOR i=2 TO r-1 STEP 2: FOR j=2
TO s-1 STEP 2: LET a(i,j)=0:
NEXT j: NEXT i
8125 CLS
8130 FOR x=1 TO r STEP 2: FOR z=0
TO 3: PLOT (x-1)*4+z,16: DRAW 0
,4*s-1: NEXT z: NEXT x
8135 FOR y=1 TO s STEP 2: FOR z=0
TO 3: PLOT 0,4*y+12+z: DRAW 4*r
-1,0: NEXT z: NEXT y
8140 RANDOMIZE : LET x=2*INT (RND*(r-
1)/2)+2: LET y=2*INT (RND*(s-1)/
2)+2: LET a(x,y)=-9: LET xl=x:
LET yl=y: LET v=1: LET max=20:
LET done=1
8145 PLOT xl*4-3,yl*4+13: DRAW 1,0:
DRAW 0,1: DRAW -1,0
8150 LET c=1
8160 LET d=INT (RND*23)+1
8165 LET vx=x+k(d(c,d)): LET vy=y+l(d
(c,d))
8170 IF vx=0 OR vx=r+1 OR vy=0 OR vy=
s+1 THEN GO TO 8200
8172 IF a(vx,vy)<>0 THEN GO TO 820
0
8175 LET v=v+1: LET a(vx,vy)=v: LET d
one=done+1: LET a(x+i(d(c,d)),y+
j(d(c,d)))=0
8177 LET xx=x+i(d(c,d)): LET yy=y+j(d
(c,d)): GO SUB 100
8178 IF done=h*w THEN GO TO 8300
8180 LET x=vx: LET y=vy: LET c=1
8185 IF v>max THEN LET max=v: LET x9
=x: LET y9=y
8190 GO TO 8160
8200 IF c<4 THEN LET c=c+1: GO TO 8
165
8210 LET v=v-1
8215 FOR i=1 TO 4: LET vx=x+k(d(i,d))
: LET vy=y+l(d(i,d)): IF vx=0
OR vy=0 OR vx=r+1 OR vy=s+1
THEN GO TO 8220
8217 IF a(vx,vy)=v THEN GO TO 8222
8220 NEXT i
8221 GO TO 8300
8222 LET x=vx: LET y=vy: GO TO 8150
8300 PLOT x9*4-3,y9*4+13: DRAW 1,0:
DRAW 0,1: DRAW -1,0
8305 COPY
8310 GO TO 10
9000 DATA 1,2,3,4,1,2,4,3,1,3,2,4,1,3
,4,2,1,4,2,3,1,4,3,2
9001 DATA 2,1,3,4,2,1,4,3,2,3,1,4,2,3
,4,1,2,4,1,3,2,4,3,1
9002 DATA 3,1,2,4,3,1,4,2,3,2,1,4,3,
2,4,1,3,4,1,2,3,4,2,1
9003 DATA 4,1,2,3,4,1,3,2,4,2,1,3,4,2
,3,1,4,3,1,2,4,3,2,1
9005 DATA 0,1,0,2,1,0,2,0,0,-1,0,-2,-
1,0,-2,0

```


Supernim

di Francesco Giunta

L'ennesimo (ma definitivo) programma per un gioco sempre divertente e stimolante

Tra i giochi matematici quello del Nim è stato sicuramente uno dei più usati per rendere "intelligenti" macchine di diverso calibro e varia natura. La possibilità di rappresentare agevolmente la strategia del Nim all'interno di un programma ha, prima o poi, coinvolto tutti coloro che si sono trovati a disposizione qualcosa di programmabile.

Raramente, però (per quanto di nostra conoscenza e per macchine "tipo Spectrum"), è stata utilizzata a pieno la potenzialità e la semplicità della strategia stessa, limitando ad esempio il numero di file di "bastoncini" (o monete) o non prevedendo la possibilità di ribaltare la regola fondamentale del gioco, possibilità che, a nostro parere, fornisce una variante interessante.

Questo programma potrebbe essere... l'ultimo, proprio perché generale. I limiti del campo di gioco, infatti, sono dettati esclusivamente dalla scelta di rappresentarlo in maniera leggibile da un lato e dall'altro di limitare "bastoncini" e file al di sotto di quantità che risulterebbero altrimenti "ingovernabili" per l'umano senza l'ausilio di carta e penna.

L'algoritmo della "mossa vincente" comunque, nella sua...freddezza e lucidità di calcolo, potrebbe trattare con la stessa comodità qualsiasi dimensione di gioco.

Rimandandovi alle altre parti per la descrizione delle variabili e del programma, resta da dire che può risultare divertente (e istruttivo) far giocare lo ZX contro se stesso: basterà, attraverso una delle opzioni del *menu* che richiedono il nome dell'umano in gioco, inserire SPECTRUM (tutto in maiuscolo). Badate bene: sarà l'unico a giocare, ma anche l'unico a vincere!

Notizie sul gioco

All'interno del programma troverete la descrizione del gioco e delle sue regole. Può esservi utile sapere che:

Descrizione del programma

Linee	Commento
10-80	dimensionamento e inizializzazione delle variabili principali
90	predisposizione del campo di gioco "standard"
100-190	stampa del <i>menu</i>
200-280	ciclo del <i>menu</i> e selezione dell'opzione
290-300	routine di "ribaltamento della regola": ad ogni passaggio la variabile <i>flag</i> assume valori 0 o 1 e r\$ "PERDE" o "VINCE"
310	inizio routine "esecuzione del gioco"
320	cancella le mosse inserite
330-340	scrivono i nomi dei giocatori e fanno lampeggiare quello del giocatore di turno
350	verifica se la partita è finita
360	verifica se muove SPECTRUM ed eventualmente salta
370-430	input del numero di riga
440-500	input del numero di bastoncini da cancellare
510-520	aggiornamento variabili principali
530-630	cancellano i bastoncini
640-670	scrivono il nome del vincitore e chiudono la routine di esecuzione del gioco ritornando al <i>menu</i>
680	inizio routine "mossa vincente"
690	in funzione della regola in vigore, valuta se saltare l'analisi dei "finali"
700-750	verificano e analizzano se la partita ha raggiunto la condizione di "finale" ed eventualmente saltano
760-800	verificano l'esistenza di una mossa vincente
810-880	calcolano la mossa vincente e ritornano per l'esecuzione della mossa stessa
890-900	in assenza di mossa vincente ricercano una mossa possibile
910-920	calcolano le mosse nei "finali" (fine della routine "mossa vincente")
930-940	routine "campo di ZX" per la scelta random del campo di gioco
950	inizializza la variabile s\$(8,4)
960-1040	routine "campo dell'umano", per l'inserimento di un campo scelto dal giocatore
1050-1080	sub-routine che chiude e conferma l'avvenuta esecuzione delle opzioni del <i>menu</i>
1090	input del nome dei giocatori
1100-1240	stampano il campo di gioco
1250-1260	sub-routine di cancellazione scritte
1270-1330	stampa copertina iniziale
1340-1460	regole e spiegazione del gioco
1470	sub-routine di pausa usata da diversi punti del programma

Descrizione delle variabili

m\$ viene inizializzata con la rappresentazione binaria dei numeri da 0 a 10

s\$(8,4)	rappresentazione binaria del numero di bastoncini per riga. La 8ª posizione viene utilizzata per i "totali" delle colonne
s(7)	numero di bastoncini per riga
n\$(2,8)	nomi dei giocatori
k\$	stringa costante utilizzata nell'input delle mosse e del campo di gioco
flag	variabile che tiene conto della regola in vigore
r\$	stringa della regola in vigore (PERDE o VINCE)
s	numero righe di bastoncini
n,x,y,g,l	variabili dei cicli FOR-NEXT
t	totali dei bastoncini rimasti (confrontata con flag indica se la partita è finita)
u,o	contano il numero di righe a 1 e a 0 (confrontando con s, si stabilisce se la partita dà al "finale")
i	punta la riga con più bastoncini nel caso di "finale"
z	utilizzata per stabilire se alcune quantità sono pari o dispari
n1,m1,i1	indici della routine "copertina"

– il gioco è impari, nel senso che, adottando una strategia corretta, vince il giocatore che muove per primo o, in funzione delle condizioni di partenza, il giocatore che muove per secondo;
– la regola fondamentale ("chi prende l'ultimo PERDE!"), anche se ribaltata,

non influisce sulla condizione della partita, ma diventa determinante nei "finali";
– nel caso di "muovere in svantaggio" non esiste una mossa di "minor danno";
– è evidente, per quanto detto, che è possibile stabilire sin dall'inizio chi può

vincere: se sapete come fare, sarebbe inutile ridirverlo e, di contro, se ancora non lo sapete, dirvelo sarebbe un peccato! Potete, comunque, fare "indagini" all'interno del programma.

Opzioni del menu

- 0 Consente all'umano di impostare il campo di gioco.
- 1 Una routine random imposta un campo di gioco casuale.
- 2 e 3 Chiedono il vostro nome, dandovi la possibilità di fare la prima e la seconda mossa.
- 4 SPECTRUM viene escluso dal gioco e vi misurate con i vostri nemici! La prima mossa al giocatore inserito per primo.
- 5 Ribalta la regola del "chi prende l'ultimo PERDE" in "chi prende l'ultimo VINCE" e viceversa.
- 6 Spiega il gioco e le regole.
- 7 Inizia la partita.

NB: Le opzioni 2,3,4 e 5 restano in vigore fino alla successiva modifica. Se dopo ogni partita non viene scelto un nuovo campo di gioco, viene riproposto un campo "standard" favorevole a chi esegue la prima mossa.

Supernim

versione per ZX Spectrum 16K/48K

```

10 REM S U P E R N I M
20 REM SPECTRUM 16/48K
30 REM F.SCO GIUNTA 1984
40 BORDER 7: PAPER 7: FLASH 0:
   CLS
50 DIM m$(11,4): DIM s$(8,4): DIM s
   (7): DIM n$(2,8): GO SUB 1270:
   LET k$="123456789tf": LET flag=
   0: LET r$="PERDE"
60 DATA "0000","0001","0010","0011"
   ,"0100","0101","0110","0111","10
   00","1001","1010"
70 FOR n=1 TO 11: READ m$(n):
   NEXT n
80 LET n$(1)="UMANO...": LET n$(2)=
   "SPECTRUM"
90 LET s=7: FOR n=1 TO 7: LET s(n)=
   n+1: NEXT n: GO SUB 950
100 CLS : INK 0: PRINT AT 0,11;
   INVERSE 1;" M E N U' "
110 INK 0: PRINT AT 4,5;"0 -->
   CAMPO DELL'UMANO"
120 INK 1: PRINT AT 6,5;"1 -->
   CAMPO DI ZX"
130 INK 2: PRINT AT 8,5;"2 -
   -> UMANO CONTRO ZX"
140 INK 3: PRINT AT 10,5;"3
   --> UMANO CONTRO UMANO"
150 INK 4: PRINT AT 12,5;"4 --
   > ZX CONTRO UMANO"
160 INK 5: PRINT AT 14,5;"5 --
   > L'ULTIMO ";r$
170 INK 0: PRINT AT 16,5;"6 --
   > SPIEGA"

```

```

180 INK 1: PRINT AT 18,5;"7 --
   > V I A !"
190 INK 0: PRINT AT 21,8; INVERSE 1;
   " SCEGLI OPZIONE ": INVERSE 0
200 IF INKEY$="6" THEN BEEP .1,30:
   GO TO 1340
210 IF INKEY$="0" THEN BEEP .1,0:
   CLS : GO TO 970
220 IF INKEY$="7" THEN BEEP .1,35:
   GO SUB 1100: GO TO 310
230 IF INKEY$="1" THEN BEEP .1,5:
   GO TO 930
240 IF INKEY$="4" THEN BEEP .1,20:
   LET h=2: GO SUB 1090: LET n$(1)
   ="SPECTRUM": GO TO 1050
250 IF INKEY$="2" THEN BEEP .1,10:
   LET h=1: GO SUB 1090: LET n$(2)
   ="SPECTRUM": GO TO 1050
260 IF INKEY$="3" THEN BEEP .1,15:
   LET h=1: GO SUB 1090: LET h=2:
   GO SUB 1090: GO TO 1050
270 IF INKEY$="5" THEN BEEP .1,25:
   GO TO 290
280 GO TO 200
290 LET flag=0↑flag: LET r$="VINCE":
   IF flag=0 THEN LET r$="PERDE"
300 GO TO 1050
310 REM ESECUZIONE DEL GIOCO
320 FOR n=1 TO 2: FLASH 0: PRINT
   AT 13,24;" ";AT 13,29;" "
330 INK 0: FLASH 2-n: PRINT AT 18,23
   ;n$(1)
340 FLASH n-1: PRINT AT 20,23;n$(2):
   FLASH 0

```

```

350 LET t=0: FOR m=1 TO s: LET t=t+s
    (m): NEXT m: IF t=0↑flag THEN
    GO TO 640
360 IF n$(n)="SPECTRUM" THEN
    GO SUB 680: GO TO 490
370 FLASH 1: PRINT AT 13,24;"R"
380 FOR x=1 TO s
390 IF INKEY$=k$(x TO x) THEN
    GO TO 420
400 IF INKEY$="a" THEN FLASH 0:
    PRINT AT 13,24;" ": GO SUB 680:
    GO TO 490
410 NEXT x: GO TO 380
420 IF s(x)=0 THEN GO TO 380
430 FLASH 0: PRINT AT 13,24;x
440 GO SUB 1470: FLASH 1: PRINT AT 1
    3,29;"Q"
450 FOR y=1 TO s(x)
460 IF INKEY$="t" THEN FLASH 0:
    PRINT AT 13,29;"t": LET y=s(x):
    LET s(x)=0: GO TO 520
470 IF INKEY$=k$(y TO y) THEN
    GO TO 500
480 NEXT y: GO TO 450
490 PRINT AT 13,24;x
500 FLASH 0: PRINT AT 13,29;k$(y
    TO y)
510 LET s(x)=s(x)-y
520 LET s$(x)=m$(s(x)+1)
530 FOR g=1 TO 100: NEXT g: FOR m=s(
    x)+y TO s(x)+1 STEP -1
540 BEEP .025,(x*m)/5-10
550 PRINT AT 1+3*(x-1),2*m;"o":
    PAUSE 2
560 PRINT AT 1+3*(x-1),2*m;"*":
    PAUSE 2
570 PRINT AT 1+3*(x-1),2*m;" "
580 BEEP .025,(x*m)/5-10
590 PRINT AT 2+3*(x-1),2*m;"o":
    PAUSE 2
600 PRINT AT 2+3*(x-1),2*m;"*":
    PAUSE 2
610 PRINT AT 2+3*(x-1),2*m;" "
620 BEEP .025,(x*m)/5-10
630 NEXT m: NEXT n: GO TO 310
640 FLASH 1: PRINT AT 11,3;"VINCE ";
    n$(3-n);" !!!"
650 PRINT AT 12,3;" PREMI UN TASTO
    ": FLASH 0
660 IF INKEY$<>" THEN GO TO 90
670 BEEP .1,(20-RND*30): GO TO 660
680 REM MOSSA VINCENTE!

690 IF flag=1 THEN GO TO 760
700 LET u=0: LET z=0: LET o=0::
    FOR m=1 TO s
710 IF s(m)=0 THEN LET o=o+1:
    GO TO 740
720 IF s(m)=1 THEN LET u=u+1: LET z
    =0↑z: GO TO 740
730 LET i=m
740 NEXT m
750 IF s=u+o+1 THEN GO TO 910

```

```

760 FOR m=1 TO 4: LET z=0: FOR l=1
    TO s
770 IF s$(1,m TO m)="1" THEN LET z=
    0↑z
780 NEXT l: LET s$(8,m TO m)=STR$ z:
    NEXT m
790 FOR m=1 TO 4: IF s$(8,m TO m)="1
    " THEN GO TO 810
800 NEXT m: GO TO 890
810 FOR l=1 TO s: IF s$(1,m TO m)="1
    " THEN GO TO 830
820 NEXT l
830 FOR g=m TO 4
840 IF s$(1,g TO g)="1" AND s$(8,g
    TO g)="1" THEN LET s$(1,g TO g
    )="0": GO TO 860
850 IF s$(1,g TO g)="0" AND s$(8,g
    TO g)="1" THEN LET s$(1,g TO g
    )="1"
860 NEXT g
870 LET z=0: FOR m=1 TO 4: LET z=z+
    VAL s$(1,m TO m)*2↑(4-m): NEXT m

880 LET x=1: LET y=s(x)-z: RETURN
890 LET x=1: FOR m=1 TO s-1: IF s(m+
    1)>s(m) THEN LET x=m+1
900 NEXT m: LET y=1+INT (RND*(s(x)-1
    )): RETURN
910 LET x=i: IF z=1 THEN LET y=s(i)
    : RETURN
920 LET y=s(i)-1: RETURN
930 REM CAMPO DI ZX
940 LET s=3+INT (RND*5): FOR n=1
    TO s: LET s(n)=1+INT (RND*9):
    NEXT n: GO SUB 950: GO TO 1050
950 FOR n=1 TO s: LET s$(n)=m$(s(n)+
    1): NEXT n: RETURN
960 REM CAMPO DELL'UMANO
970 INK 0: GO SUB 1310: PRINT AT 9,0
    ;"PREDISPONI IL TUO CAMPO DI GIO
    CO[t=riga piena]-[f se hai finit
    o]"
980 PRINT : LET s=0
990 FOR n=1 TO 11: IF INKEY$=k$(n
    TO n) THEN GO TO 1010
1000 NEXT n: GO TO 990
1010 IF n=11 THEN GO TO 1040
1020 LET s=s+1: LET s(s)=n: PRINT "Ba
    stoncini riga ";s;"....";n:
    GO SUB 1470

1030 IF s<7 THEN GO TO 990
1040 GO SUB 950: GO TO 1050
1050 CLS : INK 1
1060 PRINT AT 13,11;"{SG4}
    {G2}{SG4}{SG5}{G7}{G2}{G7}{G2}
    {SG4}{SG5}"
1070 PRINT AT 14,11;"
    {SG4}{G8}{SG4}{SG5}{G5}{G8}{G5}
    {G8}{SG1}{SG5}"
1080 GO SUB 1470: GO TO 100
1090 CLS : PRINT AT 13,7; INVERSE 1;"
    NOME DELL'UMANO ? ": INVERSE 0:

```

```

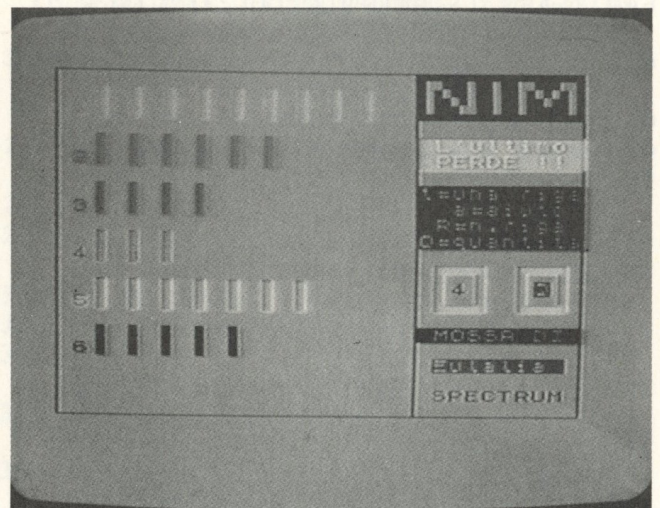
INPUT n$(h): GO SUB 1250:
RETURN
1100 REM DISEGNA IL CAMPO
1110 CLS : INK 0: PLOT 0,0: DRAW 0,17
5: DRAW 255,0: DRAW 0,-175:
DRAW -255,0
1120 PLOT 175,0: DRAW 0,175: GO SUB 1
310
1130 INK 2: INVERSE 1: PRINT AT 4,22;
" L'ultimo "
1140 INVERSE 1: PRINT AT 5,22;" ";r$;
" !! "; INK 1;AT 7,22;"t=una rig
a";AT 8,22;" a=aiuto "
1150 INK 1: PRINT AT 9,22;" R=n.riga
";AT 10,22;"Q=quantita":
INVERSE 0
1160 INK 2: PRINT AT 12,23;"{SG4}
{G3}{G7} {SG4}{G3}{G7}";AT 13,2
3;"{SG5} {G5} {SG5} {G5}";AT 14
,23;"{SG1}{SG3}{SG2} {SG1}
{SG3}{SG2}"
1170 INK 1: PRINT AT 16,22; INVERSE 1
;" MOSSA DI ": INVERSE 0
1180 FOR n=1 TO s: INK (7-n): PRINT
AT 2+3*(n-1),1;n: FOR m=1 TO s(n
)
1190 BEEP .01,(n*m)/2
1200 PRINT AT 1+3*(n-1),2*m;"{G5}"
1210 BEEP .01,(n*m)/3
1220 PRINT AT 2+3*(n-1),2*m;"{G5}"
1230 BEEP .01,40-(n*m)/2
1240 NEXT m: NEXT n: RETURN
1250 REM CANCELLA
1260 FOR g=0 TO 31: BEEP .01,g:
PRINT AT 13,g;" ": NEXT g:
RETURN
1270 REM COPERTINA
1280 LET n1=INT (RND*23): LET m1=
INT (RND*20): BEEP .04,-n1:
BEEP .02,m1: BEEP .04,n1+m1:
LET il=INT (RND*8): INK il:
IF INKEY$<>" " THEN LET m1=0:
LET n1=0: RETURN
1290 GO SUB 1300: GO TO 1270
1300 IF il=7 THEN INK 1: PRINT AT 0+
m1,22-n1; INVERSE 1;" PREMI UN "
;AT 1+m1,22-n1;"TASTO PER";AT 2
+m1,22-n1;"COMINCIARE": RETURN
1310 PRINT AT 0+m1,22-n1;"{SG4}
{G7}{SG8}{G7}{SG4}{SG8}{G3}
{SG8}{SG4}{G7}"
1320 PRINT AT 1+m1,22-n1;"{SG5}{SG1}
{G7}{G5}{SG5}{SG8}{G5}{SG6}
{SG2}{G5}"
1330 PRINT AT 2+m1,22-n1;"{SG1}{SG8}
{SG1}{SG2}{SG1}{SG8}{SG2}{2SG8}
{SG2}": RETURN
1340 CLS : GO SUB 1310
1350 PRINT AT 1,8;"Il gioco del"
1360 PRINT AT 4,0;"era gia' in voga p
resso i greci ed i cinesi."
1370 PRINT ,,"Le regole del gioco son
o molto semplici, mentre per un

```

```

a mente decimale come quella de
ll'umano e' difficile ricava
re per ogni situazione la mossa
vincente. "
1380 PRINT ,,"Per Spectrum da
lla mente binaria il NIM e' un gi
oco da ragazzi!!!"
1390 PRINT ,,"Comunque, puo'essere di
vertente provarci...dato che il
caso puo'sempre venire in aiuto!
": GO SUB 1460
1400 PRINT AT 1,8; INVERSE 1;"REGOLE
DEL GIOCO": PRINT ,,"Ognuno dei
due giocatori a turno toglie a su
o rischio e pericolo quanti bast
oncini vuole da una stessa riga
."
1410 PRINT ,,"In questo programma la
mossa va specificata indicando l
a riga e il numero di bastoncini
che si vogliono eliminare."
1420 PRINT ,,"Perde il giocatore che
toglie l'ultimo bastoncino.",:
PRINT ,,"La regola puo' essere
ribaltata,vincendo cosi' il gioc
atore che toglie l'ultimo baston
cino.": GO SUB 1460
1430 PRINT "L'opzione 5 ribalta la re
gola. (Il MENU' indica quella v
igente)"
1440 PRINT ,,"Con opzione 0 il campo
viene scelto dall'umano."
: PRINT ,,"Con opzione 1 ci pens
a Spectrum .....con l'aiuto
del caso!"
1450 PRINT ,,"LIMITI DEL CAMPO DI
GIOCO": PRINT ,,"- N. di righe:
7 ": PRINT "- N. di bastoncini
per riga: 10": GO SUB 1460:
GO TO 100
1460 FLASH 1: PRINT AT 21,8;"PREMI UN
TASTO": PAUSE 0: FLASH 0: CLS :
RETURN
1470 FOR t=0 TO 10: BEEP .01,(10+
INT (RND*20)): NEXT t: RETURN

```



Boomerang War

di Carlo Cappelli

È nato il figlio di "Space Invaders" e "Breakout": un incrocio interessante e impegnativo per lo ZX81

In questo gioco dovete difendervi dall'invasione delle schiere nemiche che scendono inesorabilmente verso di voi; per contrattaccare avete tre cannoni che sparano bombe boomerang: rimbalzando, esse tornano verso di voi e possono anche distruggervi, se non sarete più che abili a spostarvi.

Il vostro contrassegno è il "+" e si trova nella parte bassa dello schermo; per spostarlo usate i tasti "5" e "7", mentre per sparare verso destra usate il tasto "8" e verso sinistra il tasto "9". Le bombe boomerang sono inarrestabili e indistruttibili; se colpiscono le pareti o il nemico, vengono riflesse di 90°, mentre se si urtano tra di loro frontalmente, ritornano indietro.

Nel caso in cui alcune di esse siano in rotta di collisione con voi, e non siate in grado di muovervi, potete difendervi sparandone delle altre e sfruttando i rimbalzi favorevoli.

Per bloccare l'avanzata del nemico dovete distruggere completamente la prima fila (quella più vicina a voi); ricordate che avete a disposizione "solo" 40 bombe boomerang per distruggere una fila. Se ne sparate poche, difficilmente riuscirete a bloccare l'avanzata nemica, mentre se ne sparate tante, è facile che veniate colpiti. Se il nemico scende fino



al vostro livello, siete spacciati: quindi fuoco a volontà, ma con accortezza. Si possono selezionare undici livelli di gioco (da 0 a 10); col crescere dei livelli aumenterà la velocità sia delle bombe che delle file nemiche.

Per quanto riguarda l'eventuale trascrizione del programma inseritelo e salvatelo su cassetta. Date ora RUN: se non

ci sono errori (in caso contrario vi sarà segnalata la linea da modificare), il programma trasformerà tutte le REM in una sola unica linea e farà partire il gioco. Date BREAK e salvate nuovamente su cassetta una copia del programma, sicuramente esatta e partente immediatamente, da usarsi abitualmente.

Boomerang war versione per ZX81 16K

```

2 REM 3A007EA7C832027E3E00 4F
3 REM 32047E3A027EA7171721 3D
4 REM 057E16005F1922B44022 3A
5 REM BE40232322AE4022D240 5A
6 REM 1121002A137E22BB402A 38
7 REM 117E36004440CD204122 43
8 REM 117E3610FE15CA90413A 60
9 REM 047EA7C4EA402ABBB4022 6F
10 REM 137E3A027E3DC28740C9 62
11 REM 010707E1C9FF2003E1E1 76
12 REM C932067E2A0C40110E00 35
13 REM 197E3CFE26280277C936 5B
    
```

```

14 REM 1C2B18F37EFE88200532 5E
15 REM 047E3600FE1A00D1C923 6F
16 REM ED52CDFE4021574122BB 7C
17 REM 4060691923CDFE402120 5B
18 REM 4122BB406069ED522B0C 6F
19 REM FE40210D4122BB406069 5B
20 REM ED523CDFE40213C4122 6A
21 REM 8B406069192BCDFE4021 6B
22 REM 204122BB406069ED522B 5E
23 REM CDFE4021574122BB4060 6C
24 REM 691923CDFE40210D4122 53
25 REM 8B406069ED5223CDFE40 84
26 REM 213C4122BB406069192B 47
27 REM CDFE4021204122BB4060 67
28 REM 69ED522BCDFE406069C9 8B
29 REM 01FF7DE1C906022A077E 63
    
```

```

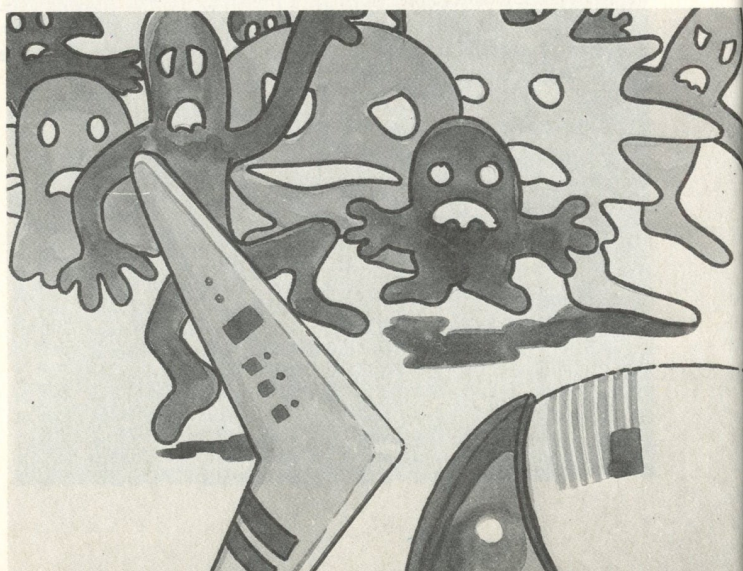
30 REM 3E000C8004FE619FE1136 5E
31 REM 00200123FE0920012E7E 35
32 REM FE3020032A077E361922 46
33 REM 077E100B79E6074F00C5 64
34 REM 444DFE05201E3A007E30 47
35 REM FE1E284532007EA71717 4E
36 REM 16005F21057E19712370 38
37 REM 2336202336410179FE03 54
38 REM 20233A007E3CFE1E2822 42
39 REM 32007EA7171716005F2A 36
40 REM 077E444D21057E197123 30
41 REM 70233600233641CD8240 4C
42 REM 180400E118F706FF3E1A 57
43 REM 000000003D020FA10F53A 42
44 REM 3440FEF838292A057EE5 6F
45 REM 061F7EFE8828062310F8 59
46 REM C3DC40E1E51121001922 68
47 REM 057EEBE1011F00ED003A 6D
48 REM 017E3DC832017EC39541 61
49 REM 00000000000000000000 00
50 REM 00000000000000000000 00
55 IF PEEK 17004=255 THEN GOTO
500
50 FAST
70 LET K=16514
80 FOR I=1 TO 49
85 IF PEEK (K-3)<>25 THEN GOTO
200
90 LET A=0
100 FOR J=1 TO 10
110 LET L1=PEEK K-28
120 LET L2=PEEK (K+1)-28
130 LET A=A+L1*16+L2
140 LET K=K+2
150 NEXT J
160 LET L1=PEEK (K+1)-28
170 LET L2=PEEK (K+2)-28
180 LET A=INT (A/10)
190 IF A=L1*16+L2 THEN GOTO 230
200 PRINT "LA REM NUMERO ";I;"E
SBAGLIATA"
210 PRINT "CORREGGILA E DAI DI
NUOVO IL RUN"
220 STOP
230 LET K=K+9
240 NEXT I
250 LET K=16514
260 LET H=K
270 POKE 16512,2
280 POKE 16511,6
290 FOR I=1 TO 49
300 FOR J=1 TO 10
310 LET L1=PEEK K-28
320 LET L2=PEEK (K+1)-28
330 POKE H,L1*16+L2
340 LET K=K+2
350 LET H=H+1
360 NEXT J
370 LET K=K+9
380 NEXT I
390 POKE 17004,255
400 SLOW
500 POKE 16389,126
510 CLEAR
520 PRINT "*****"
*****
530 PRINT "BOOMERANG U
AR *****"
540 PRINT "*****"
*****
545 LET V=1
550 LET DF=256*PEEK 16397+PEEK
16396+1
550 PRINT
555 GOSUB 1200
570 PRINT "LIVELLO (0-10) ?"
580 INPUT L
590 IF L>10 OR L<0 THEN GOTO 58
0
600 CLS
610 PRINT " SCORE 000000"
620 LET A$="INT ((DF+33*A+B)/25
5)"
630 LET B$="DF+33*A+B-VAL A$*25
6"
640 LET A=19
650 LET B=15
660 POKE 32263,VAL B$

```

```

670 POKE 32264,VAL A$
680 LET A=2
690 LET B=1
700 GOTO 1000
710 LET K=USR 16789
720 IF PEEK 32257=1 THEN GOTO 8
20
730 IF K=1799 THEN GOTO 800
740 LET V=V+1
750 IF V=4 THEN GOTO 820
760 PRINT AT 10,2;"PRIMO" AND (
V=2);"SECONDO" AND (V=3);"CANNO
NE DISTRUTTO"
770 FOR I=1 TO 50
780 NEXT I
790 GOTO 1000
800 LET L=L+.4*(L<15)
802 PRINT AT 10,4;"SCHIERA DIST
RUTTA"
805 FOR I=1 TO 50
807 NEXT I
810 GOTO 1000
820 PRINT AT 10,13;"GAME OVER"
830 PRINT AT 13,2;"VUOI RICOMIN
CIARE (SI-NO)"
840 INPUT C$
850 IF C$="SI" THEN CLS
860 IF C$="SI" THEN GOTO 500
870 STOP
1000 POKE 32255,0
1010 POKE 32257,16
1020 POKE 32261,VAL B$
1030 POKE 32262,VAL A$
1040 POKE 16923,27-INT L
1050 POKE 16937,250-2*INT L
1060 PRINT AT 1,0;"
1070 FOR I=1 TO 19
1080 PRINT "
1090 NEXT I
1100 PRINT "
1110 PRINT AT 2,1;"
1120 GOTO 710
1200 PRINT "GLI INVASORI INCALZA
NO. DEVI BLOCCARE LA LORO AVA
NZATA SPARANDO BOMBE BOOME
RANG, NE HAIA DISPOSIZIONE 40. L
E BOMBE RIMBALZANDO TI POSSO
NO COLPIRE E HAI SOLO 3 CANNONI
"
1210 PRINT "PER BLOCCARE IL NEMI
CO DEVI DISTRUGGERE LA SUA P
RIMA SCHIERA."
1220 PRINT "PER SPOSTARTI A SINI
STRA USA IL TASTO 5"
1230 PRINT "PER SPOSTARTI A DEST
RA USA IL TASTO 7"
1240 PRINT "PER SPARARE I TASTI
8 E 9"
1250 RETURN

```



Totoriduttore

di Massimo Serio

Il modo più comodo e veloce di ridurre un sistema

Questo programma consente la riduzione di un sistema integrale fino a un massimo di 2304 colonne (corrispondente a 2 triple e 8 doppie).

L'uso della routine è molto semplice: basta inserire il sistema nella sua forma consueta e le specifiche richieste per la riduzione. L'eliminazione delle colonne avviene confrontando la quantità dei segni 1-X-2 di ognuna di esse e la quantità dei segni voluta ed immessa in precedenza nelle specifiche richieste dal computer.

La routine prevede anche l'uso (facoltativo) di una stampante; ciò è opportuno, in quanto i tempi di elaborazione e riduzione del sistema sono un po' lunghi e la stampa su carta avviene automaticamente ad ogni videata completa con l'istruzione COPY.

ESEMPIO DI STAMPA:
(stampante GP-50 S)

PROGRAMMA DI RIDUZIONE DEI SISTEMI INTEGRALI.					
1	1X	INTEGRALE col.	144		
3	1X		L. 50400		
4	1X2	risse n.7			
5	1X2	doppie n.4.	triple n.2		
6	1X2	RIDOTTO col.	50		
8	1X2		L. 17500		
7	1X2	Specifiche per la riduzione:			
9	1	SEGN1: min. max.			
10	1				
11	1	-1-	n.7	n.3	n.5
12	1X	-X-	n.10	n.3	n.7
13	1X	-2-	n.4	n.1	n.8

```

XXXX 11XX 1XX1  XX11  XX1X  111X
XXXX  XXXX  XXXX  XXXX  XXXX  XXXX
XX1  X21  XX1X  X1X2  21XX  1211

XX2X  XXX2  22XX  XXXX  X222  XX22
2222  XXXX  XXXX  XXXX  XXXX  XXXX

111X  XXXX  XX22  2111  1111  XXXX
XXXX  XXXX  XXXX  XXXX  XXXX  XXXX
1111  1111  1111  1111  1111  1111

1111  1111  1111  1111  1111  1111
XXXX  XXXX  XXXX  XXXX  XXXX  XXXX
1111  1111  1111  1111  1111  1111
1111  1111  1111  1111  1111  1111
    
```

Output del programma su stampante



Descrizione del programma

Linee	Commento
10-120	intestazione; predispongono il programma all'uso eventuale della stampante; memorizzano il costo attuale di una colonna (variabili s\$a\$-c)
130-340	immissione e memorizzazione del sistema integrale nella variabile p\$(13,3); eventuali correzioni
350-500	visualizzano un quadro riassuntivo del sistema introdotto
510-780	inserimento delle quantità minima e massima dei tre segni 1-X-2 ed eventuali correzioni (var. z(6))
790-1030	sviluppo integrale del sistema (var. c\$(n,13)) e calcolo del numero di colonne valide in base alle quantità inserite
1040-1190	visualizzazione ed eventuale stampa delle sole colonne valide
1200-1240	eventuale replay e fine
1250-1320	subroutine richiamate da più punti del programma

Totoriduttore
versione per ZX Sepctrum 48K

```

10 REM -----TOTORIDUTTORE-----
    -----Massimo Serio-----
    -----Napoli---1984-----
20 LET j=0: GO SUB 1300: BEEP .05,3
   0: BEEP .05,20
30 PRINT AT 13,1;"Premi un tasto pe
   r cominciare.": PAUSE 0: BEEP .0
   5,30: BEEP .05,20
40 PRINT AT 13,1;"
   ": INPUT "Quanto co
   sta una colonna? L.": LINE a$
50 IF a$="" THEN GO TO 40
60 FOR a=1 TO LEN a$: IF a$(a)<"0"
   OR a$(a)>"9" THEN GO TO 40
70 NEXT a: LET c=VAL a$: IF c<350
   OR c>1000 THEN GO TO 40
80 BEEP .05,30: BEEP .05,20:
   INPUT "Usi la stampante <S/N>?
   ": LINE s$: IF s$="" THEN
   GO TO 80
90 LET s$=s$(1): IF s$<>"s" AND s
   $<>"S" THEN GO TO 130
100 PRINT AT 11,0;"Se la stampante n
   on e' collegata-spegnere il comp
   uter- e colle-gare la stampante
   ."
110 GO SUB 1320
120 PRINT AT 15,0;"In caso contrario
   ""premere un tasto per continua
   re.": PAUSE 0: GO SUB 1300
130 DIM p$(13,3): PRINT AT 3,0;"Inse
   rire i pronostici."
140 PLOT 115,135: DRAW 0,-127:
   PLOT 96,108: DRAW 47,0: PLOT 96
   ,76: DRAW 47,0: PLOT 96,44:
   DRAW 47,0
150 LET r=4: FOR a=1 TO 13: LET r=r+
   1+(1 AND a=4 OR a=7 OR a=10)
160 PRINT AT r,14-LEN STR$ a;a
170 BEEP .01,30: INPUT "Pron. N. ";(
   a), LINE p$(a)
180 IF p$(a)=" " OR p$(a,1)=" "
   THEN GO TO 170
190 FOR b=1 TO 3: IF p$(a,b)<>"1"
   AND p$(a,b)<>"x" AND p$(a,b)
   <>"X" AND p$(a,b)<>"2" AND p
   $(a,b)<>" " THEN GO TO 170
200 IF p$(a,b)="x" THEN LET p$(a,b)
   ="X"
210 NEXT b
220 PRINT AT r,15;p$(a): NEXT a
230 GO SUB 1250
240 PRINT AT 5,0;"Fisse ";f;AT 6,0;
   "Doppie ";d;AT 7,0;"Triple ";t;
   AT 9,0;"Col.":col;AT 11,0;"L.":l
250 IF col>2304 THEN PRINT #0;AT 0,
   8;"Troppe colonne!": BEEP .01,30
   : BEEP .01,20: PAUSE 0: RUN

```

```

260 PRINT #0;"Vuoi effettuare correz
   ioni? S/N": BEEP .01,30
270 PAUSE 0: LET z$=INKEY$: BEEP .01
   ,30: IF z$<>"s" AND z$<>"S"
   THEN GO TO 350
280 INPUT "Pron. da correggere N. ";
   a: IF a<1 OR a>13 THEN GO TO 28
   0
290 LET r=a+(4 AND a>=1 AND a<=3)+(5
   AND a>=4 AND a<=6)+(6 AND a>=7
   AND a<=9)+(7 AND a>=10 AND a<14
   ): PRINT AT r,11; FLASH 1;">"
300 BEEP .01,30: INPUT "Pronostico?"
   , LINE p$(a)
310 FOR b=1 TO 3: IF p$(a,b)<>"1"
   AND p$(a,b)<>"x" AND p$(a,b)
   <>"X" AND p$(a,b)<>"2" AND p
   $(a,b)<>" " THEN GO TO 300
320 IF p$(a,b)="x" THEN LET p$(a,b)
   ="X"
330 NEXT b
340 PRINT AT 5,0;" ";AT 6,0
   ";" ;AT 7,0;" ";AT 11,0;"
   ";AT 9,0;" ";AT r,11;" ";AT r,15;
   p$(a): GO TO 230
350 GO SUB 1300: PLOT 6,153: DRAW 24
   3,0: DRAW 0,-131: DRAW -243,0:
   DRAW 0,131: PLOT 28,149: DRAW 0
   ,-124: PLOT 17,123: DRAW 38,0:
   PLOT 17,91: DRAW 38,0: PLOT 17,
   59: DRAW 38,0: PLOT 69,22:
   DRAW 0,68: DRAW 180,0
360 PRINT AT 3,9;"INTEGRALE col.":
   AT 4,21;"L.":AT 5,9;"fisse n.":
   AT 6,9;"doppie n. triple n.":
   AT 8,9;"RIDOTTO col.":AT 9,21;
   "L.":AT 11,9;"Specifiche per":
   AT 12,9;"la riduzione :":AT 14,9
   ;"SEGNI: min. max.":AT 16,
   9;"-1- n. n. n.":AT 17,9
   ;"-X- n. n. n.":AT 18,9;
   "-2- n. n. n."
370 LET r=2: FOR a=1 TO 13: LET r=r+
   1+(1 AND a=4 OR a=7 OR a=10)
380 PRINT AT r,3-LEN STR$ a;a;AT r,4
   ;p$(a): BEEP .005,40: NEXT a
390 PRINT AT 3,24;col;AT 4,24;l;AT 5
   ,18;f;AT 6,18;d;AT 6,30;t
400 LET uno=0: LET ics=0: LET due=0:
   FOR a=1 TO 13
410 IF p$(a,1)="1" OR p$(a,2)="1"
   OR p$(a,3)="1" THEN LET uno=un
   o+1
420 IF p$(a,1)="X" OR p$(a,2)="X"
   OR p$(a,3)="X" THEN LET ics=ic
   s+1
430 IF p$(a,1)="2" OR p$(a,2)="2"
   OR p$(a,3)="2" THEN LET due=du
   e+1
440 NEXT a

```



```

450 PRINT AT 16,16;uno;AT 17,16;ics;
    AT 18,16;due: IF j=1 THEN
        GO TO 530
460 PRINT AT 20,0;"Vuoi ridurre il s
    istema <S/N?": BEEP .05,30:
        BEEP .05,20
470 IF INKEY$="s" OR INKEY$="S"
    THEN GO TO 510
480 IF INKEY$<>"n" AND INKEY$<>"
    N" THEN GO TO 470
490 PRINT AT 20,0; BRIGHT 1;"
    BUONA FORTUNA! " :
        BEEP .05,30: BEEP .05,20: IF s$
    ="s" OR s$="S" THEN COPY
500 STOP
510 BEEP .05,30: BEEP .05,20:
    PRINT AT 20,0;"Inserisci le qua
    ntita' min e max";AT 21,2;"per o
    gni segno."
520 DIM z(6)
530 FOR a=1 TO 6
540 IF j=1 THEN GO TO 650
550 IF a=1 THEN INPUT "Segno -1- mi
    n. "; LINE a$
560 IF a=2 THEN INPUT "Segno -1- ma
    x. "; LINE a$
570 IF a=3 THEN INPUT "Segno -X- mi
    n. "; LINE a$
580 IF a=4 THEN INPUT "Segno -X- ma
    x. "; LINE a$
590 IF a=5 THEN INPUT "Segno -2- mi
    n. "; LINE a$
600 IF a=6 THEN INPUT "Segno -2- ma
    x. "; LINE a$
610 IF a$="" THEN GO TO 550
620 FOR r=1 TO LEN a$: IF CODE a$(r)
    <48 OR CODE a$(r)>57 THEN
        GO TO 550
630 NEXT r: LET z(a)=VAL a$
640 IF z(a)<0 OR z(a)>13 THEN
        GO TO 550
650 IF a=1 THEN PRINT AT 16,22;" "
    ;AT 16,22;z(a)
660 IF a=2 THEN PRINT AT 16,29;" "
    ;AT 16,29;z(a)
670 IF a=3 THEN PRINT AT 17,22;" "
    ;AT 17,22;z(a)
680 IF a=4 THEN PRINT AT 17,29;" "
    ;AT 17,29;z(a)
690 IF a=5 THEN PRINT AT 18,22;" "
    ;AT 18,22;z(a)
700 IF a=6 THEN PRINT AT 18,29;" "
    ;AT 18,29;z(a)
710 NEXT a
720 IF j=1 THEN GO TO 970
730 PRINT AT 21,2;" " ;
    AT 20,0; BRIGHT 1;" " ESAT
    TO <S/N?": BEEP .05,3
    0: BEEP .05,20
740 IF INKEY$="s" OR INKEY$="S"
    THEN BEEP .05,30: BEEP .05,20:
        GO TO 770
750 IF INKEY$="n" OR INKEY$="N"
    THEN GO TO 510

```

```

760 GO TO 740
770 LET mi=INT (col*1.48/60): LET se
    =INT (col*1.48-mi*60)
780 PRINT AT 21,4; BRIGHT 1;"min. ";
    mi;" e "; "sec. ";se;" circa.";
    AT 20,0; FLASH 1;" " ATT
    ENDERE! " "
790 DIM c$(col,13)
800 FOR a=1 TO 13: IF p$(a,2)<>" "
    THEN LET r=1: GO TO 820
810 FOR b=1 TO col: LET c$(b,a)=p$(a
    ): NEXT b: NEXT a
820 FOR e=a TO 13: IF p$(e,3)<>" "
    THEN LET s=3: LET r=r*3: LET r
    i=r/s: GO TO 850
830 IF p$(e,2)<>" " THEN LET s=2:
    LET r=r*2: LET ri=r/s: GO TO 85
    0
840 FOR b=1 TO col: LET c$(b,e)=p$(e
    ): NEXT b: GO TO 890
850 LET p=1: LET v=0: LET q=1
860 LET c$(p,e)=p$(e,q): LET v=v+1:
    IF v=ri THEN LET q=q+1: LET v=
    0
870 IF q>s THEN LET q=1
880 LET p=p+1: IF p<=col THEN
    GO TO 860
890 NEXT e
900 LET el=0: FOR a=1 TO col: LET un
    =0: LET du=0: LET ic=0: FOR b=1
    TO 13
910 IF c$(a,b)="1" THEN LET un=un+1
920 IF c$(a,b)="X" THEN LET ic=ic+1
930 IF c$(a,b)="2" THEN LET du=du+1
940 NEXT b
950 IF un<z(1) OR un>z(2) OR ic<z(3)
    OR ic>z(4) OR du<z(5) OR du>z(6
    ) THEN LET el=el+1
960 NEXT a
970 PRINT AT 8,24;col-el;AT 9,24;(co
    l-el)*c: IF j=1 THEN PRINT AT 2
    0,9; BRIGHT 1;"premi un tasto";
    AT 20,26; FLASH 1;"REPLAY":
        BEEP .05,30: BEEP .05,20:
            PAUSE 0: GO TO 1030
980 PRINT AT 21,4;" " ;
    " ;AT 20,0; BRIGHT 1;" "
        VA BENE <S/N?": " "
990 GO SUB 1320
000 IF INKEY$="s" OR INKEY$="S"
    THEN GO TO 1030
010 IF INKEY$<>"n" AND INKEY$<>"
    N" THEN GO TO 1000
020 GO TO 350
030 PRINT AT 20,0;" " : IF s$="s" OR s
    $="S" THEN COPY
1040 CLS : PRINT #0;AT 0,4;"colonne e
    liminate n.0";AT 1,4;"colonne st
    ampate n.0": IF j=1 THEN

```

```

PRINT AT 0,0; BRIGHT 1; FLASH 1
;"REPLAY"
1050 LET el=0: LET co=1: FOR a=1 TO c
ol: LET un=0: LET ic=0: LET du=0
: FOR b=1 TO 13
1060 IF c$(a,b)="1" THEN LET un=un+1
1070 IF c$(a,b)="X" THEN LET ic=ic+1
1080 IF c$(a,b)="2" THEN LET du=du+1
1090 NEXT b: IF un<z(1) OR un>z(2)
OR ic<z(3) OR ic>z(4) OR du<z(5)
) OR du>z(6) THEN LET el=el+1:
GO TO 1180
1100 LET r=3: FOR b=1 TO 13: IF b=4
OR b=7 OR b=10 THEN LET r=r+1
1110 PRINT AT r,co; BRIGHT 1;c$(a,b):
LET r=r+1: NEXT b: PRINT #0;
AT 1,24;a-el
1120 LET co=co+1: IF co=5 OR co=10
OR co=21 OR co=26 THEN LET co=
co+1
1130 IF co=15 THEN LET co=17
1140 IF co<31 THEN GO TO 1180
1150 IF s$="s" OR s$="S" THEN COPY :
LET co=1: GO TO 1170
1160 PRINT AT 20,9; BRIGHT 1;"premi u
n tasto": BEEP .05,30: BEEP .05,
20: LET co=1: PAUSE 0
1170 FOR b=20 TO 3 STEP -1: PRINT
AT b,1;"
": NEXT b
1180 BEEP .005,30: PRINT #0;AT 0,24;e
l: NEXT a

```

```

1190 IF (col-el)/24<>INT ((col-el)/
24) AND s$="s" OR s$="S" THEN
COPY
1200 PRINT AT 20,6; BRIGHT 1;"premi
per rivedere";AT 21,6;"premi
per finire.";AT 20,12;
INVERSE 1;"R";AT 21,12;"F"
1210 IF INKEY$="r" OR INKEY$="R"
THEN BEEP .05,30: BEEP .05,20:
GO TO 1240
1220 IF INKEY$="f" OR INKEY$="F"
THEN CLS : PRINT AT 11,0;
BRIGHT 1;" BUONA FORTUN
A! ": BEEP .05,30:
BEEP .05,20: STOP
1230 GO TO 1210
1240 LET s$="n": LET j=1: GO TO 350
1250 LET f=0: LET d=0: LET t=0: LET c
ol=1: FOR a=1 TO 13
1260 IF p$(a,2 TO 3)=" " THEN LET n
=1: LET f=f+1: GO TO 1290
1270 IF p$(a,3)=" " THEN LET n=2:
LET d=d+1: GO TO 1290
1280 LET n=3: LET t=t+1
1290 LET col=col*n: NEXT a: LET l=c*c
ol: RETURN
1300 CLS : PLOT 0,175: DRAW 255,0:
DRAW 0,-15: DRAW -255,0: DRAW 0
,15
1310 PRINT AT 0,0; BRIGHT 1; OVER 1;"
PROGRAMMA DI RIDUZIONE
DEI SISTEMI INTEGRALI.
": RETURN
1320 FOR a=1 TO 20: BEEP .05,0:
BEEP .05,30: NEXT a: RETURN

```



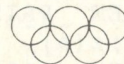
Totocalcio

"AL SERVIZIO DELLO SPORT"

PARTITE DEL 22-9-85

squadra 1 ^a	squadra 2 ^a	1	2	3	4
1 Bari	Roma	① ● ②	① X ●	① ● ②	① ● ②
2 Fiorentina	Milan	● X ②	● X ②	● X ②	● X ②
3 Inter	Avellino	● X ②	● X ②	● X ②	● X ②
4 Juventus	Pisa	● X ②	● X ②	● X ②	● X ②
5 Lecce	Torino	① X ●	① ● ②	① X ●	① ● ②
6 Napoli	Atalanta	● X ②	● X ②	● X ②	● X ②
7 Udinese	Sampdoria	① X ●	① ● ②	① ● ②	① X ●
8 Verona	Como	① ● ②	① X ●	① X ●	① X ●
9 Cagliari	Genoa	① X ●	① X ●	① X ●	① X ●
10 Perugia	Ascoli	① ● ②	① ● ②	① ● ②	① ● ②
11 Triestina	Monza	① ● ②	① ● ②	① ● ②	① ● ②
12 Trento	Parma	● X ②	● X ②	● X ②	● X ②
13 Cosenza	Salernitana	① ● ②	① ● ②	① ● ②	① ● ②

MINISTERO FINANZE... coni



1	2	3	4	1	2	3	4

Insert

I GIOCHI VIRGIN IN EDICOLA

Grazie ad un accordo con la Jackson e la Virgin Games Italiana, tutti i videogiochi di maggior successo della casa inglese saranno disponibili in edicola.

Seguendo una formula già sperimentata con successo per la serie Jackson Soft Oro, ogni mese verrà inclusa in una nuova rivista una compilation di tre videogiochi originali. Tutti i giochi saranno venduti sulla cassetta originale proveniente dall'Inghilterra. Sarete sicuri di non avere problemi nel caricamento o di non trovare spiacevoli bug nel programma.

Strangeloop, Sorcery, Falcon Patrol 1 e 2, Golf, Racing Manager, Ghost Town, Space Command, The Island, The Biz, sono alcuni tra i titoli che saranno inseriti in ogni numero della rivista, con un abbinamento vario ed assortito, affiancando per esempio ad un adventure un videogioco spaziale oppure ad uno di strategia uno di guerra e così via.

Nella rivista, oltre a notizie e curiosità, non mancheranno consigli e trucchi per aiutarvi a raggiungere i risultati migliori.

SOFTWARE FRESCO FRESCO

Molte novità in arrivo per tutti i possessori dello Spectrum.

Iniziamo da **FOURTH PROTOCOL**, destinato a diventare un best seller. Gioco del tipo adventure dinamico, in **FOURTH PROTOCOL** dovete scoprire un traditore e il piano per destabilizzare un paese occidentale.

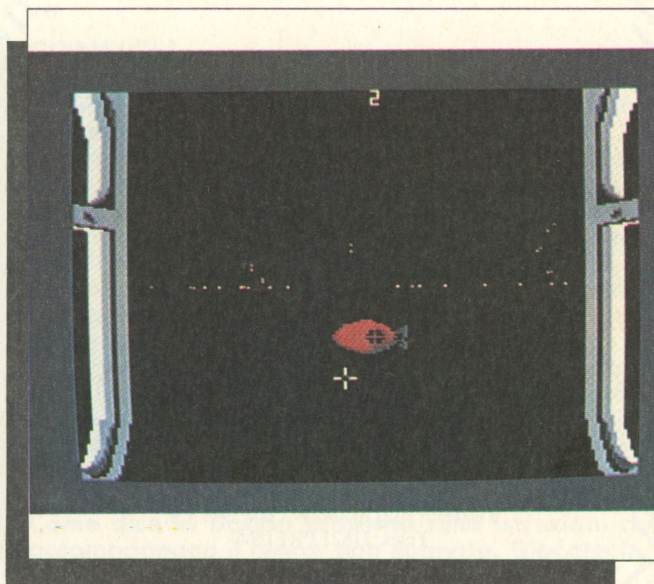
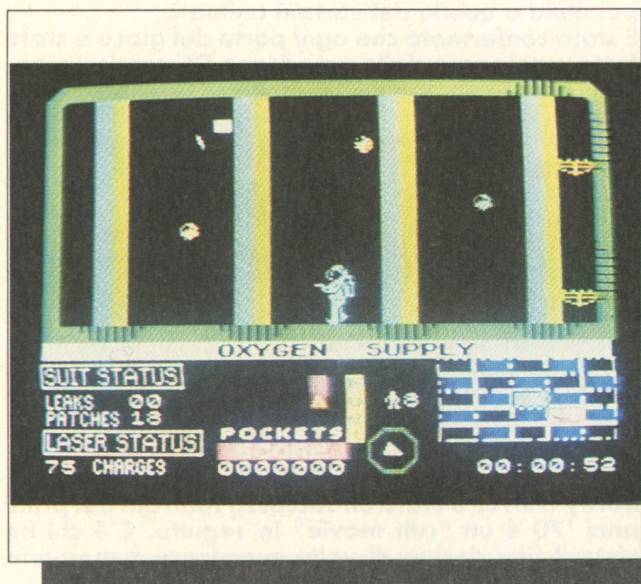
Una serie di immediati e chiari schermi con icone vi aiutano nel gioco.

Nel panorama del software Spectrum gli sport continuano comunque ad essere i veri protagonisti.

Ispirato all'arcade Karate Champ è **THE WAY OF THE EXPLODING FIST**, della Melbourne House, con ben 18 tipi di colpi, parate o salti.

La Imagine ha realizzato una serie di giochi per conto della casa giapponese Konami, specializzata in videogiochi da bar. Vedremo al più presto discipline come nuoto, tiro con l'arco, tiro al piattello, salto triplo, sollevamento pesi e il volteggio al cavallo.

Gli altri giochi della Konami che saranno realizzati per lo Spectrum sono **TENNIS**, **KUNG FU**, **HYPER RALLY** e il **GOLF**.



La boxe sta avendo un boom notevole. I migliori giochi sembrano essere PUNCH OUT della Elite e ROCKY della Gremlins. Ambedue utilizzano la visione posteriore, come nel gioco da bar. PUNCH OUT, tra l'altro, incorpora i dati di 8 eccentrici pugili che Frank Bruno, lo sfidante per il titolo mondiale, deve incontrare. Il terzo gioco di boxe è KNOCK OUT della Alligata, per un solo giocatore, con una prospettiva meno spettacolare.

Daley Thompson, il simpatico ed eccentrico campione olimpico di decathlon, torna sui vostri schermi, sempre grazie alla Ocean. Tiro alla fune, tiro con la pistola, calci di rigore, tuffi, canottaggio, ciclismo, slalom gigante e salto con gli sci sono le discipline che mettono alla prova la vostra abilità con quella dell'asso inglese in DALEY THOMPSON'S SUPERTEST.

Sempre la Ocean ha appena presentato NEVERENDING STORY, ispirato al film fantastico "La storia infinita".

La US Gold, invece, ha realizzato anche per lo Spectrum DAMBUSTERS che rievoca ad una ardita azione della II Guerra Mondiale, e BUCK ROGERS.

Jet Set Willy invece una nuova avventura più eccitante, JET SET WILLY II. Questa volta non dovrà più pulire la casa, ma risolvere puzzle e combattere alieni.

Tra i giochi di simulazione, la Digital Integration ha in programma TOMAHAWK, un simulatore di volo per elicottero con grafica 3D, e TT RACER, una simulazione di corsa di moto alla guida di una Suzuki Project 500.

Infine JUMP JET, un simulatore di volo per un jet a decollo verticale, realizzata dalla Anirog.

ARRIVA SUPERMAN

Disponibile nei negozi è il nuovo videogioco, che ha come protagonista SUPERMAN.

La Beyond, dopo numerose proposte ricevute dalla casa americana First Star, ha accettato la proposta di realizzare un gioco con il famoso eroe, già protagonista di fumetti, cartoni animati e film.

Il compito di realizzare il gioco è stato affidato alla Monolith, una nuova label legata alla Beyond, che ha promesso di utilizzare una grafica il più simile possibile a quella dei cartoni animati.

È stato confermato che ogni parte del gioco è stata fatta omologare dalla americana DC comics, creatrice del personaggio di SUPERMAN.

Non è escluso che altri personaggi della DC comics, come Batman e Supergirl, vengano utilizzati in altre avventure.

THE ROCKY HORROR SHOW

Computer: Spectrum 48K

Prodotto da : CRL

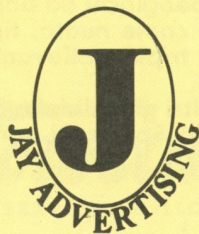
Distribuito da: Lago

Supporto: cassetta

Prezzo: L. 25.000

Rocky Horror è stato un successo teatrale dei primi anni '70 e un "cult movie" in seguito. C'è chi ha visto il film decine di volte e conosce a memoria canzoni, musica e personaggi.

Concessionaria Pubblicità



Agenzia per le tre venezie

Pier Filippo OBBER

Via Nazionale, 70

38070 IMER (TN)

tel. 0439/67221

J. ADVERTISING s.r.l.

Sede Legale, Direzione e Amministrazione:

Viale F. Restelli, 5 - 20124 Milano

Telefono (02) 6880606 - 6085941 r.a.

Telex 316213 REINA

Dopo il caricamento del programma vi viene propinata la simpatica danza (con tanto di parole) della canzone e al termine una schermata che vi mostra il tetro castello in cui si svolgerà l'azione del gioco. Il gioco è fedelmente ispirato alla storia, e premendo il tasto ENTER appaiono Brad o Janet, i protagonisti, e vi viene chiesto chi dei due desiderate impersonare.

I COMANDI

Segue un menu con tutte le opzioni di controllo del game che spaziano dalla tastiera al joystick Sinclair.

Per effettuare la scelta si preme il tasto SPACE, e l'ENTER quando la periferica che abbiamo deciso di usare lampeggia.

Con la tastiera il movimento del protagonista è assegnato ai tasti A e Z per i movimenti verticali e N e M per quelli orizzontali.

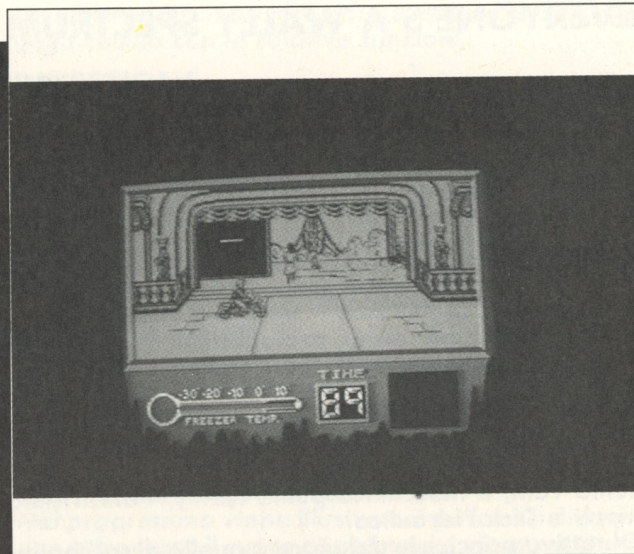
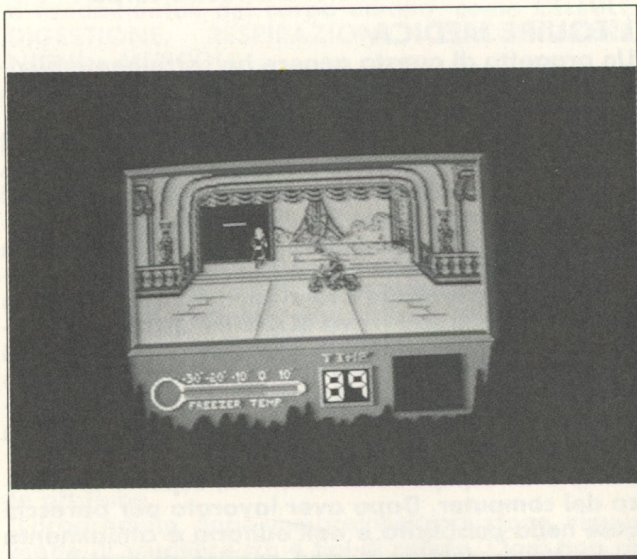
Premendo lo SPACE, o il pulsante di fuoco se si usa il joystick, si raccolgono gli oggetti o le chiavi sparsi per tutto il castello.

OBIETTIVO

Il nostro dovere è quello di salvare il partner caduto nelle perfide mani del Dr. FRANK, che sta cercando, per mezzo della diabolica macchina chiamata MEDUSA, di pietrificarlo.

La macchina che può salvare il nostro compagno da questa triste fine si chiama DEMEDUSA. Il cattivello e perverso Dr. FRANK ha naturalmente ben pensato di distruggerla, spargendone i pezzi nella tetra casa.

Cercando nel castello troveremo quindi tutti i pezzi che ci servono: dovremo raccogliergli e portarli, ma solo uno per volta, dietro ad un sipario, dove assembleremo la macchina.



Per entrare in una determinata stanza occorre prima aprire la porta, chiusa naturalmente a chiave. Di chiavi ne esistono comunque a sufficienza, anch'esse, come detto, sparse per la casa.

Nella parte bassa dello schermo è visualizzato un termometro che indica la temperatura, un orologio che scandisce il tempo in cui dovremo salvare il partner e un riquadro che mostra le parti della Demedusa raccolte.

Appena raccolto un pezzo, questo lampeggerà fino a che lo avremo riposto dietro al sipario.

Sempre nella parte bassa dello schermo viene indicato se siamo in possesso o meno di qualche chiave.

CONSIGLI

Il gioco non è un gran che come grafica e forse un po' noioso, visti i continui spostamenti che siamo costretti a fare per assemblare la macchina.

Anche la difficoltà non è eccessiva, se non in alcuni punti dove occorre un buon tempismo.

Non abbiate timore di usare il vecchio ascensore presente nel castello, dato che ci fa risparmiare un po' di tempo. Attenti, oltre alle visibili insidie del piano superiore, anche al Dr. Frank che gira armato. La vita a nostra disposizione è una soltanto, teniamola cara.

Durante il gioco incontrerete anche altri personaggi come EDDIE, il "ciccione" motorizzato, tutti collaboratori del malefico dottore, che vi diranno frasi apparentemente senza senso.

Non sono comunque pericolosi, anche se è bene stargli alla larga.

Come dice lo slogan presente sulle istruzioni che accompagnano il gioco, non sognate. Giocateci! E... occhio ai vestiti, aggiungo io.

EVERYONE'S A WALLY SPECTRUM

Computer: Spectrum 48K
Prodotto da: Mikro-gen
Distribuito: Jackson Soft
Supporto: cassetta
Prezzo: L. 10.000 in edicola con rivista

Il quarto numero di ORO SOFT, il software che vi gasa e che ogni mese vi propone in edicola – direttamente dall'Inghilterra – un gioco originale e di prima scelta, ha come protagonista il simpatico Wally Week in una delle sue strampalate avventure.

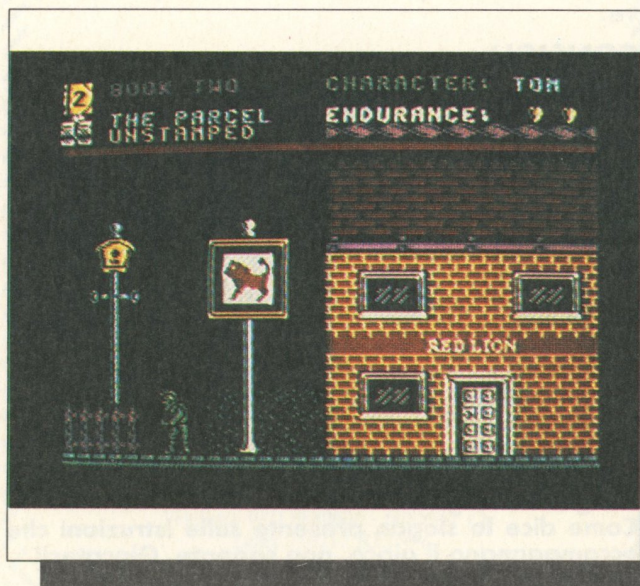
In EVERYONE'S A WALLY abbiamo l'occasione di conoscere Wilma e Herbert, rispettivamente moglie e figlio di Wally, e alcuni eccentrici personaggi come Tom, il meccanico punk, Harry, l'elettricista hippy e Dick l'idraulico.

Obiettivo principale del gioco è quello di raccogliere le lettere del codice richiesto per aprire la cassaforte della banca e pagare gli stipendi di tutti i personaggi.

Le lettere sono sparse assieme a numerosi oggetti per tutta la città.

Ognuno dei componenti della gang ha dei lavori particolari da svolgere e l'aumento delle paghe vi procura the, pranzi e pause che, oltre ad aumentare la vostra energia, indicano anche il progresso nell'avventura.

La complessità del gioco non ha uguali, visto che la novità principale è la possibilità di poter comandare a scelta uno dei 5 personaggi. In questo modo potete svolgere lavori che altrimenti Wally non riuscirebbe a realizzare ed inoltre vi quadruplica le vi-



te.

Ogni personaggio è interattivo e può essere comandato premendo il corrispondente tasto quando si trova sullo schermo.

I lavori da compiere sono dei più vari, come aggiustare la fontana, costruire un muro, riparare il gas, completare una nave e così via.

Gli schermi del gioco sono numerosi e corrispondono a classici luoghi della città, come strade, parchi, birrerie, zoo, drogherie, uffici postali, cantieri navali e cantieri edili, tanto per citarne alcuni.

Come in PYJAMARAMA, ogni oggetto ha il suo uso particolare e la sua funzione: tocca a voi scoprire quali, per riuscire a risolvere questo gioco e fare la felicità della gang.

BODY WORKS

Computer: Spectrum 48K
Prodotto da: Genesis
Supporto: cassetta
Prezzo: L. 45.000

Avete mai pensato di utilizzare il computer per studiare l'anatomia?

Qualche studente starà già scuotendo la testa, pensando alle ore e ore trascorse sui libri o davanti a qualche tabella, nel tentativo di imparare i nomi di ossa, tessuti ed organi.

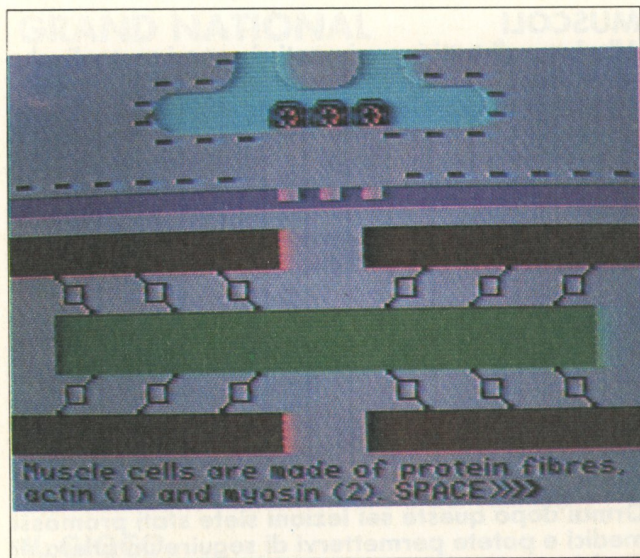
Niente paura, perché BODY WORKS vi accompagnerà in un viaggio fantastico senza tormentarvi troppo, anzi si propone di aiutarvi ad imparare con divertimento come funziona il vostro corpo.

L'EQUIPE MEDICA

Un progetto di questo genere ha certamente bisogno di veri esperti per la sua realizzazione: ecco qualche notizia sui creatori di BODY WORKS.

Il Dr. Miller, che si è laureato in scienze naturali al St. John's College di Cambridge ed è diventato dottore in medicina all'Università di Londra nel 1959, ha sempre avuto il pallino della comunicazione. Non a caso è conosciuto anche in America, non solo come scienziato, ma anche come scrittore, attore e regista teatrale e televisivo, e come tale cura la trasmissione televisiva BODY IN QUESTION, sulla storia della medicina e suoi sono i libri illustrati THE HUMAN BODY. Ha aiutato questo vulcanico personaggio il Dr. David Jefferys, proveniente dall'Università di Londra e diventato dottore dopo un tirocinio al Guy Hospital nel 1976.

Completa l'equipe John Cameron, esperto di grafica del computer. Dopo aver lavorato per parecchi anni nella pubblicità e nell'editoria è attualmente il direttore creativo di una società di consulenza londinese.



LA CONFEZIONE

Scopo di BODY WORKS è illustrare i principali processi fisiologici che forniscono energia e costruiscono i materiali fondamentali per la sopravvivenza e la crescita dell'organismo.

Un'elegante confezione colorata, delle dimensioni di un libro, contiene le due cassette con sette programmi, una tavola del corpo umano scientificamente esatta e un preciso e dettagliato manuale, che, in linea con lo scopo del programma, approfondisce i vari argomenti.

Dei sette programmi, sei riguardano funzioni e parti fondamentali del corpo umano, come CELLULE, DIGESTIONE, RESPIRAZIONE, CIRCOLAZIONE, NERVI e MUSCOLI.

Il settimo programma è la classica prova finale, dove, simulando una maratona, viene verificato se avete imparato le sei lezioni.

Tutti i programmi possono essere trasferiti su microdrive.

LA CELLULA

"Il corpo umano è una confederazione di cellule individuali. Ogni cellula è una parte del corpo e così ogni parte del corpo è una collezione di organismi che formano la vita del pianeta terra."

In questo modo i Dr. Miller e Jefferys introducono sul manualetto il primo programma dedicato alle cellule.

Il paragrafo prosegue analizzando l'ambiente, l'energia, la struttura, la digestione e la sintesi delle proteine.

Il programma, dopo una breve introduzione, vi permette di scegliere tra 4 tipi di cellule. Ogni tipo ha una breve spiegazione di carattere generale, men-

tre muovendo un cursore potete vedere i particolari della cellula con le relative funzioni.

DIGESTIONE

L'apparato digerente permette all'organismo di rendere assorbibili le sostanze proteiche contenute nei cibi.

Il manuale illustra l'apparato digerente, dalla bocca, attraverso l'esofago, fino all'intestino.

Il programma vi mostra come avviene la digestione di protidi, lipidi e glucidi, dei carboidrati e come viene controllato l'ingresso del cibo.

LA RESPIRAZIONE

L'ossigeno è un elemento fondamentale per il nostro metabolismo. Grazie ad esso il nostro corpo produce energia, ossidando il glucosio.

Naturalmente, l'ossigeno viene introdotto nell'organismo attraverso l'apparato respiratorio e trasportato dal sangue ai vari tessuti.

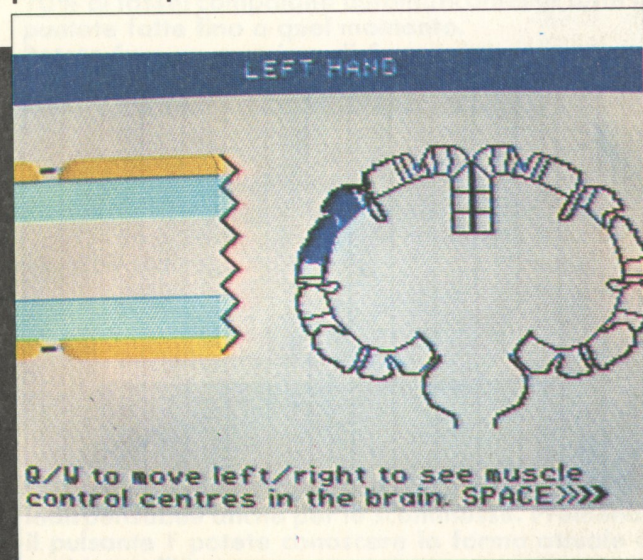
Nel programma viene illustrato il percorso del sangue attraverso il cuore e i polmoni e viene mostrata la simulazione della respirazione, compresi gli effetti durante la corsa o quando si trattiene il respiro.

CIRCOLAZIONE

Il sangue, pompato dal cuore, trasporta il nutrimento e l'ossigeno per cellule e tessuti attraverso le vene e le arterie. In questo programma viene illustrato il percorso del sangue nel cuore e nei polmoni, nella parte superiore e in quella inferiore del corpo, nei reni e nell'apparato digerente.

NERVI

Come tutti sanno, il sistema nervoso ha la funzione di trasmettere informazioni dalle cellule in tutte le parti del corpo fino al cervello.

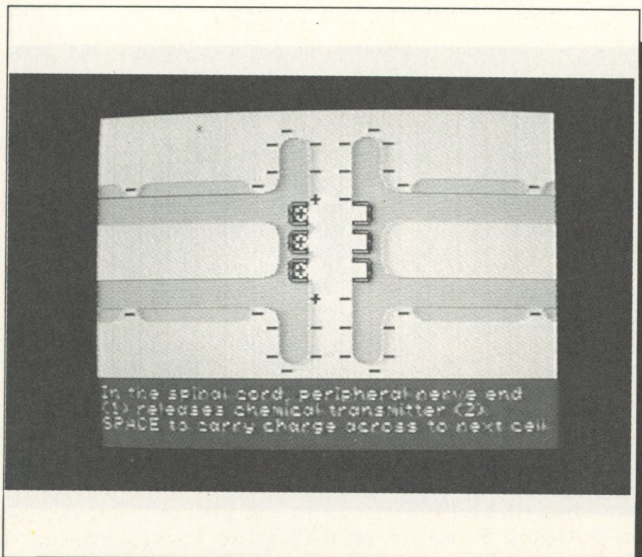
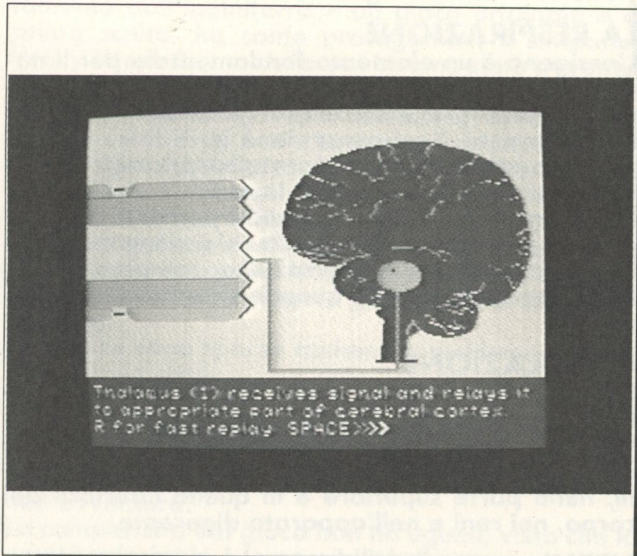


Insert

Sensazioni come il piacere, il dolore, la luce, la temperatura sono stimoli trasmessi sotto forma di impulsi elettrici attraverso i nervi.

Altri nervi hanno il compito di trasmettere le eventuali reazioni dal cervello ai muscoli.

Il programma mostra come i nervi trasmettono gli impulsi al cervello e contiene un test della vista e del suono, per controllare come vengono inviati e interpretati differenti segnali al cervello.



MUSCOLI

Alla base di tutti i movimenti ci sono i muscoli, che sono divisi in due gruppi: volontari, come quelli che muovono gambe, braccia, ecc., ed involontari, come quelli del cuore e dei visceri.

Quest'ultimo programma vi mostra come il cervello controlla i muscoli e come questi si contraggono. Infine, con un videogame a due livelli potete imparare, giocando, come funziona il controllo dei muscoli.

MARATONA

Il programma finale è la simulazione di una maratona lunga da 5 a 26 miglia.

Ormai dopo queste sei lezioni siete stati promossi medici e potete permettervi di seguire un atleta di qualsiasi età nel duro sforzo di correre una maratona.

Dopo aver scelto l'età, il sesso e la forma, la mole, e altre caratteristiche dell'atleta iniziate la prova. Grazie alle sofisticate apparecchiature elettroniche, potete controllare la velocità, la distanza percorsa e tutte le funzioni fisiologiche dell'atleta. Sulla sinistra c'è la sua immagine presa dall'alto. Al termine, se riuscirete a far terminare la gara senza nessun infarto, saprete anche la media tenuta.

Un grafico tridimensionale vi indica il ritmo della respirazione, la distribuzione di ossigeno, i battiti cardiaci al minuto e i litri di sangue pompati.

Inoltre potete conoscere la temperatura del corpo, la quantità di sangue pompato ad ogni battito, la sudorazione e la quantità di acido lattico prodotto.

Tramite i tasti T e H potete passare dalla visione del terreno, per sapere se è pianeggiante o in salita, a quella dell'elettrocardiogramma.

Se riuscirete a far correre a una buona media tutti i soggetti senza causare attacchi cardiaci, potrete sperare nella carriera di preparatore atletico.

CONCLUSIONI

BODY WORKS, più che un classico videogioco, è uno di quei programmi intelligenti che, utilizzando il computer, riescono a far avvicinare un vasto pubblico ad argomenti importanti, anche se impegnativi.

Non a caso il programma è stato realizzato da due esperti in medicina. Inoltre l'esauriente manuale e la tavola a colori rendono BODY WORKS ancora più completo.

GRAND NATIONAL

Computer: Spectrum 48K
Prodotto da: Elite
Distribuito da: Lago
Supporto: cassetta
Prezzo: L. 25.000

La casa produttrice Elite ha realizzato una linea di giochi sportivi dalla grafica e dalla struttura standard.

GRAND NATIONAL è il primo della serie ed è fedelmente ispirato a una delle corse di galoppo più famose d'Inghilterra.

Il clima che si respira in ogni sala corse e all'ippodromo prima di una gara viene riproposto in questo gioco, che vi dà la possibilità di impersonare il doppio ruolo dello scommettitore e del fantino.

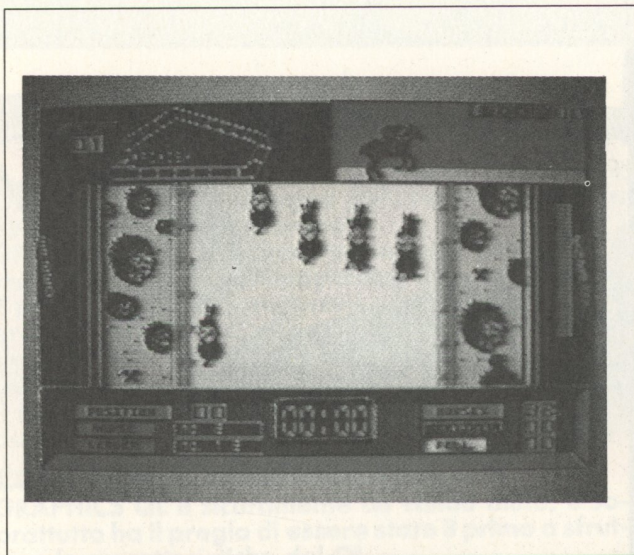
IL GIOCO

La corsa sulla quale si basa il gioco è il Grand National, il Gran Premio siepi che si svolge ogni anno all'Aintree Race vicino a Liverpool. Questo durissimo gran premio, della lunghezza di due giri e con un totale di 30 ostacoli da superare, ammette la partecipazione di un massimo di 40 cavalli.

Nella realtà la corsa ha una durata di 10 minuti, mentre sul vostro computer è più breve, soprattutto se non riuscirete a tagliare il traguardo. Il gioco inizia con l'edizione 1985 del Grand National.

NOVITÀ

Novità assoluta per i giochi ispirati alle corse dei cavalli è la vostra partecipazione al Gran Premio da protagonista.



Prima della competizione vera e propria bisogna però espletare alcune operazioni preliminari. Lo schermo iniziale vi fornisce lo stato del terreno, il nome del cavallo favorito e l'elenco di tutti i cavalli a disposizione.

LE SCOMMESSE

All'inizio avete una somma di 1000 sterline, disponibile per le scommesse. Premendo il tasto B a fianco dei cavalli, potete fare la puntata sul cavallo preferito. Dopo aver deciso il numero e la somma da puntare (ricordatevi che vi viene trattenuto il 10% di tasse) compaiono automaticamente tutte le puntate fatte fino a quel momento.

Potete fare un massimo di 5 scommesse per gara e non vengono accettate scommesse la cui vincita paghi più di 9999 sterline.

Difficilmente un cavallo dato sfavorito dai bookmaker riesce a vincere un Grand National, a meno che siate voi a cavalcarlo.

Attenzione a non rimanere in bolletta, perché altrimenti non potrete gareggiare nemmeno con il ronzino più scalcagnato.

LA SCELTA

Investiti i vostri denari, dovete scegliere il cavallo con cui partecipare alla corsa.

Un numero in reverse vi segnala i 10 cavalli che potete ingaggiare.

Sul retro delle istruzioni è riportato un utile elenco di tutti i cavalli, con le loro caratteristiche principali e il tipo di terreno preferito. Questo elenco vi è indispensabile anche per le scommesse. Premendo il pulsante F potete conoscere la forma attuale di tutti i cavalli.

Naturalmente, conviene scegliere il cavallo che si

Insert

adatta meglio alle condizioni del terreno ed impostare la gara secondo le sue caratteristiche. Come detto, potete anche rischiare di scegliere un cavallo dato sfavorito dai bookmaker. Con una gara accorta potrete portarlo alla vittoria, guadagnando così un bel gruzzolo con le scommesse.

LA GARA

Questa è la fase più impegnativa del gioco: riuscire a terminare la corsa senza cadere rovinosamente su qualche siepe è molto difficile.

Al centro dello schermo c'è la pista vista dall'alto con i cavalli pronti alla partenza.

In alto a sinistra una piantina vi indica lo sviluppo del percorso con la disposizione delle siepi.

Una piccola finestrella a destra vi mostra il cavallo lateralmente durante la corsa, aiutandovi quando dovete saltare la siepe.

Ai due lati dello schermo due barre vi indicano la velocità e l'energia.

Altri dati riguardano la vostra posizione, il numero del vostro cavallo, il numero del leader della corsa, i cavalli in gara, quelli ritirati e quelli caduti.

Con i tasti Z e X determinate la direzione del cavallo, con O incrementate la sua velocità e con P effettuate il salto.

Fondamentale è la scelta del tempo per quest'ultima operazione. Un errore vi costerà il capitombolo e la conseguente eliminazione dalla corsa.

Evitate assolutamente di essere stretti contro la staccionata o tra due cavalli: perderete velocità e non riuscirete più a saltare.

È possibile che al termine della gara ci sia un intervento dei giudici per punire eventuali scorrettezze, modificando così il risultato finale.

FINE CORSA

Al termine della gara appare lo schermo con la classifica finale.

Tutti i cavalli, compresi i ritirati e i non partenti, vengono elencati con a fianco le loro quotazioni.

In alto vengono riportate le vostre eventuali vincite, comprendenti il premio corsa (40000 sterline per il primo, 14.000 al secondo, 7.000 al terzo e 3.000 per il quarto posto) e le eventuali vincite nelle scommesse. Le multe ricevute dai giudici per comportamento scorretto verranno sottratte dalla vostra somma.

Finito di consultare il risultato finale affrontate un altro anno e un'altra corsa, premendo semplicemente il tasto S.

CONCLUSIONI

Il gioco ha un solo piccolo difetto, se così possiamo chiamarlo: l'estrema difficoltà nel riuscire a terminare la gara.

Due giri con trenta siepi da superare, correndo tra più di trenta puledri scatenati, sono veramente duri da completare. D'altronde anche il Gran National reale si riduce sempre a una vera gara ad elimina-

zione.

Comunque, se non riuscite a terminare la corsa, potete sempre sperare di avere scommesso sul cavallo giusto.

GRAPHIC QL

Computer: QL

Prodotto da: Talent

Distribuito da: Lago

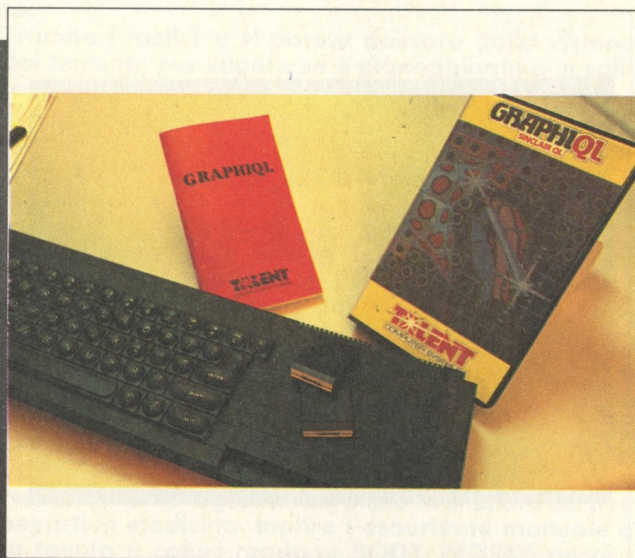
Supporto: microdrive

Prezzo: L. 99.000

Malgrado l'ancora statica produzione di software per il QL, la Talent, ditta inglese già specializzata nella produzione di software grafico, ha anticipato tutti, realizzando questo pacchetto per lo sfruttamento delle notevoli capacità grafiche del QL.

La confezione è molto curata e comprende un manuale chiaro e ricco di suggerimenti (interessante anche la sezione dedicata ai consigli per fotografare il video) e due microcartucce con il programma principale e tre schermi dimostrativi.

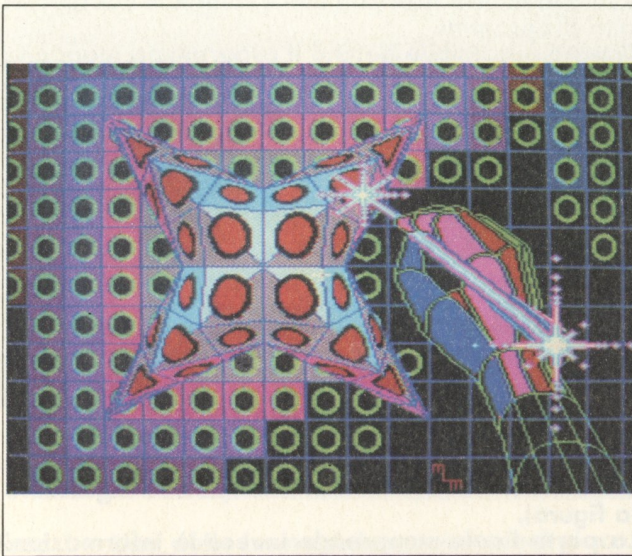
L'altissima definizione del computer, di 256 x 256 pixel, è sfruttata al massimo, e per disegnare si può scegliere tra utilizzare il joystick o la tastiera. Il disegno può essere definito nei minimi particolari, visto che si può intervenire su ogni singolo pixel, accendendolo oppure colorandolo anche con il comando di ZOOM, che ingrandisce qualsiasi zona dello schermo. Altre sono le possibilità di editing



che rendono molto versatile questo programma, come il poter definire intere sezioni del disegno con il comando BLOCK CURSOR, per poi poterle ruotare, spostare e copiare, oppure introdurre caratteri e simboli di varia dimensione in aree precedentemente definite.

Interessante è anche la possibilità di definire un'area esterna a quella del disegno, nella quale si possono creare fino a ventotto figure senza rovinare quella originale.

Ciascuna delle figure può essere richiamata in un secondo tempo con un tasto e quindi inserita nel disegno. I comandi principali prevedono la possibilità di tracciare cerchi, ellissi, raggi e linee, mentre con il comando PAINT BRUSH si può disegnare a mano libera e con l'AIR BRUSH si ha l'effetto "grafiti newyorkesi", come se coloraste con una bomboletta spray.



Ovviamente, ogni disegno può essere salvato e quindi ricaricato da microdrive.

GRAPHICS QL, naturalmente, non è un programma studiato per creare immagini fini a se stesse. È possibile infatti introdurre i propri disegni come programmi in BASIC o in ASSEMBLER.

Una speciale routine, facilmente modificabile per adattarla ai vari tipi di periferiche compatibili con il QL, permette di stampare il disegno, tutto o in parte.

CONCLUSIONI

GRAPHICS QL è sicuramente un valido aiuto, e soprattutto ha il pregio di essere stato il primo a sfruttare le caratteristiche del QL.

La Talent ha annunciato, per la gioia di tutti i "quel-

listi", la prossima realizzazione di ANIMATE e BACKDROP, due nuovi programmi per l'animazione e la creazione di disegni miniaturizzati, che dovrebbero completare questo package grafico.

THE LOST KINGDOM OF ZKUL

Computer: QL
Prodotto da: Talent
Distributore: Lago
Supporto: microdrive
Prezzo: L. 65.000

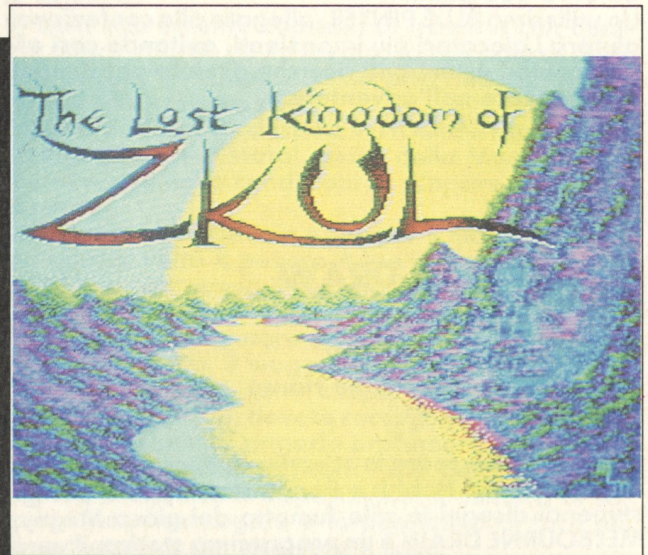
Per quanto ne sappiamo, THE LOST KINGDOM OF ZKUL è uno dei primi adventure realizzati per il QL. Più di 200 locazioni, indovinelli e un interprete che capisce centinaia di parole sono alcune delle caratteristiche di questo adventure di solo testo.

LA STORIA

Il luogo dove è ambientata l'avventura è la città di DORNED, situata sulla montagna di FORSAKEN e abitata da un popolo di nani. Dorned era una città pacifica e felice, almeno fino a quando non venne attaccata dagli abitanti della lontana CARACAS.

La battaglia fu durissima e i nani, grazie anche all'aiuto del mago TARADON, ebbero il sopravvento, anche se pagando un alto prezzo in vite umane. La vostra avventura ha inizio molti anni dopo, e ha come scopo il ritrovamento del tesoro custodito in quel luogo ormai abbandonato.

Vi aiuta in questa difficile impresa il vostro amico



ELDOMIR.

Durante il viaggio, tra mille insidie dovete raccogliere oggetti, tesori e le informazioni indispensabili per entrare nella capanna.

Vi imparerete in molti personaggi: alcuni vi daranno utili consigli, altri faranno di tutto per mettervi i bastoni fra le ruote.

Tuttavia non disperate, perché potete sempre sconfiggerli in un duello all'ultimo sangue.

I COMANDI

Una caratteristica interessante di ZKUL è l'interattività del computer con il giocatore. Per esempio, se siete in difficoltà, il vostro accompagnatore vi aiuterà, dandovi consigli in cambio di punti.

Molti sono i comandi classici per muoversi nelle varie locazioni, combattere, raccogliere o lasciare gli oggetti, dialogare con i personaggi.

Attenzione, perché molte locazioni sono raggiungibili con uno stesso comando, in quanto comunicanti per vie curve.

Comandi speciali come HEALTH vi sono indispensabili prima di ogni scontro, per sapere la vostra forza ed evitare brutte sconfitte, oppure SCORE, per avere il punteggio e il numero di mosse realizzate fino a quel momento.

Il punteggio viene determinato dagli oggetti raccolti e dalla difficoltà dei luoghi visitati.

Ogni volta che insultate il computer o rimanete uccisi perdetevi dei punti.

Avete a disposizione 3 vite, e con meno di 50 punti non potete proseguire.

Il punteggio finale può variare da un minimo di 50 ad oltre 600 punti.

Un utilissimo BLUE PINTER, allegato alla confezione, aiuterà i giocatori più impazienti, evitando così affannose ed estenuanti ricerche di aiuto per riuscire ad avanzare nell'avventura.

MELBOURNE DRAW

Computer: Spectrum 48K
Prodotto da: Melbourne House
Supporto: cassetta

Utilizzato per creare le schermate di presentazione di giochi come Hobbit e Penetrator, oppure per gli stupendi disegni in stile fumetto del gioco Mugsy, MELBOURNE DRAW è un programma storico, il vero capostipite di tutti i programmi grafici per lo Spectrum.

Il cursore utilizzato per disegnare viene comandato in 8 direzioni da altrettanti tasti.

I comandi a disposizione sono molti e permettono di posizionare il cursore (SKIP), disegnare (SET), cancellare (RESET), invertire (INVERT) e muovere (SCROLL).

Potete aggiungere anche del testo alle vostre figure, utilizzando un set di caratteri creato appositamente.

Non manca anche un comando fondamentale come FILL, utilizzato per colorare parti di schermo contornate. La lentezza di questa operazione non deve spaventarvi, perché può risultarvi utile se, dopo aver commesso qualche errore, volete interromperla per non rovinare il lavoro.

Il programma principale, contenente tutte le informazioni per disegnare, è realizzato in linguaggio macchina.

Tutti i disegni possono essere salvati su nastro ed eventualmente ricaricati con il comando LOAD"" in altri programmi.

Questo è possibile perché il programma mantiene sempre due copie del disegno in memoria: la prima è quella mostrata sullo schermo; la seconda è localizzata all'indirizzo 32768, è lunga 6912 byte ed è quella che può essere salvata.

Oltre ai disegni, è possibile caricare i caratteri d'uso modificati in precedenza.

LO SCHERMO DI LAVORO

Mentre lavorate sullo schermo una linea vi tiene informati sui vari comandi usati.

Il primo messaggio contiene il modo in cui state lavorando: SKIP, SET, RESET, INVERT, SCROLL o TESTO. Il secondo vi indica se avete selezionato lo schermo o gli attributi. Il terzo contiene le coordinate X e Y, dove si trova il cursore. Un quadrato nella parte destra del video vi indica in quale parte dello schermo vi trovate (molto utile quando ingrandite la figura).

La parte finale comprende invece le informazioni sui parametri degli attributi.

Le dimensioni dello schermo sono di 24 righe per 32 colonne. Ogni riga e colonna è formata da un quadrato di 8 pixel. Avete due modi per operare: con l'opzione SCREEN potete disegnare, stabilendo quale pixel deve essere acceso e quale spento; la seconda, chiamata ATTRIBUTES, è quella che definisce i colori nel disegno.

Un'opzione particolare vi permette di scegliere tra pulire il fondo (paper) o i contorni (ink) dagli attributi, FLASH e BRIGHT compresi, oppure lo schermo senza avere effetto sugli attributi, oppure cancellare tutto il disegno.

La possibilità di ingrandire lo schermo di 4 e 16 volte è un valido aiuto, anche per agevolarvi nel disegnare con la non eccezionale definizione dello Spectrum.

Altre funzioni interessanti sono la possibilità di scroll (muovere il disegno) e quella di richiamare

uno sfondo a griglia come riferimento nel disegnare.

Lavorando sugli attributi, l'unità di misura passa dal singolo pixel al carattere e viene utilizzato un megacursore.

CONCLUSIONI

Semplice e completo, MELBOURNE DRAW rimane un programma utilissimo per realizzare i vostri sogni artistici. Inoltre, le immagini create possono diventare doppiamente utili e completare i vostri programmi con schermate di caricamento o immagini particolari.

Alberto Rossetti

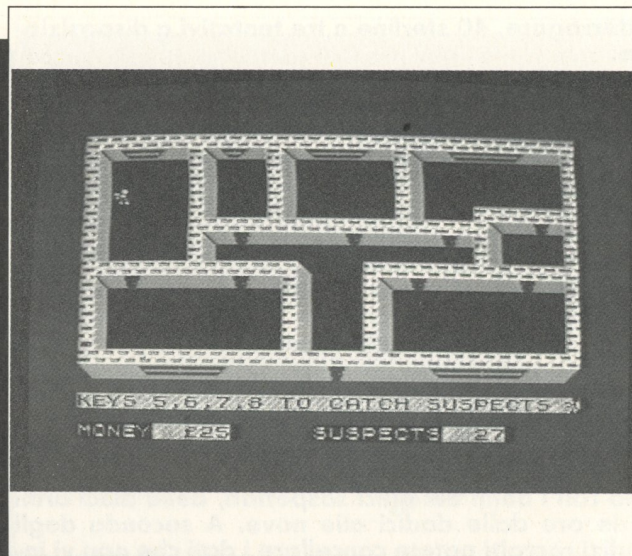


Foto 1 - Una delle immagini contenute nel gioco MUGSY e realizzate con il MELBOURNE DRAW.

WHODUNIT

Computer: Spectrum 48K
 Prodotto da: Mastertronic
 Distribuito da: Mastertronic
 Supporto: cassetta
 Prezzo: L. 8.000

Ispirato da molti giochi di società, WHODUNIT vi pone di fronte a un assassino.



Se avete il pallino della cronaca nera e del mistero, non siete più obbligati a frequentare i posti più malfamati, oppure ad intraprendere la carriera di ispettore di polizia.

Certo, non vivrete le emozioni di Marlowe, non avrete una moglie comprensiva come quella di Maigret e, peggio ancora, non potrete gustare piatti prelibati come fanno Nero Wolfe o Poirot. Potrete comunque ugualmente applicare la vostra cultura giallesca, vogliamo sperare formatasi solo sui libri o vedendo film, per il compito meno rischioso di risolvere i casi di WHODUNIT.

IL GIOCO

Vi sarà sicuramente capitato di trovarvi con qualche amico intorno ad un tavolo, con un paio di dadi, per giocare a "Cluedo" o ad altri "board game" ispirati alla cronaca nera.

Proprio a "Cluedo" si ispira WHODUNIT (più o meno vuol dire chi è stato), gioco della Mastertronic, che vi mette nelle condizioni di scoprire chi è l'assassino.

Solitamente il colpevole è sempre il maggiordomo, ma questa volta il povero è la vittima, visto che il gioco non contempla il suicidio.

L'indagine si svolge tra le quattro mura di una casa. Oltre a scoprire l'autore dell'assassinio, dovete individuare l'arma, il luogo e l'ora del delitto.

Tutti i personaggi sono sospettati e, come ogni buon investigatore, dovete raccogliere informazioni anche a costo di pagarle profumatamente.

Ci sono tre livelli di difficoltà, che determinano la somma in sterline a disposizione, il numero di sospettati da interrogare e la possibilità di fare più di un tentativo per risolvere il caso.

Il livello più facile è il terzo, con 70 sospettati da

Insert

interrogare, 40 sterline e tre tentativi a disposizione.

Scelto il livello di difficoltà, si entra nel vivo del gioco.

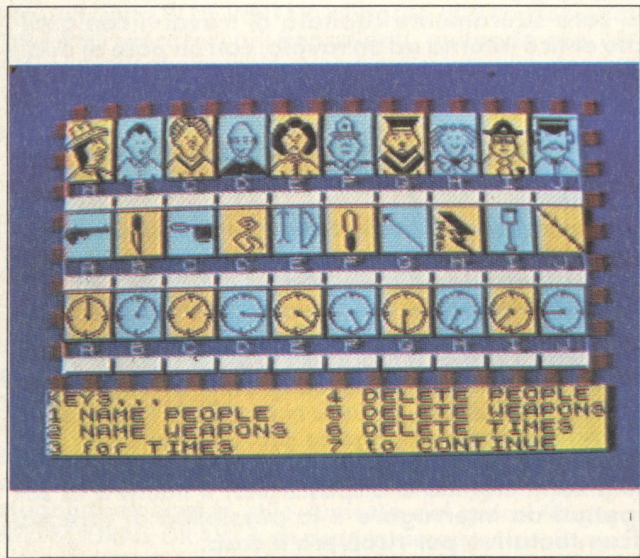
La prima fase è in perfetto stile arcade-labirinto. Sullo schermo avete un appartamento visto dall'alto. Usando i tasti delle frecce del vostro Spectrum, dovete muovere velocemente il detective tra le varie stanze, raggiungendo, quando appare, il sospettato per interrogarlo.

Non fatevelo sfuggire, perché perderete un possibile indizio ed il numero dei sospettati da interrogare diminuirà. Più impegnativo è raggiungere il sospettato al livello più difficile. Dopo aver pagato la somma di una sterlina (la si può anche risparmiare, rinunciando però ad avere le informazioni) e aver letto sullo schermo la risposta, riportate i dati ricevuti sui vostri appunti, che appaiono sullo schermo con i nomi dei dieci sospettati, delle dieci armi e le ore dalle dodici alle nove. A seconda degli indizi raccolti potete cancellare i dati che non vi interessano.

Quando vi sentite in grado di formulare una ipotesi provate ad indovinare assassino, arma, ora e luogo del delitto. Ricordatevi che una volta terminati i soldi, e i sospettati da interrogare, non avete più diritto ad alcun tentativo.

Al termine vi viene assegnato un punteggio, che tiene conto del numero dei sospettati interrogati, dei soldi risparmiati e del numero degli elementi indovinati.

Naturalmente, se siete riusciti a risolvere interamente il caso, avrete un punteggio alto e riceverete anche gratificanti complimenti.



CONCLUSIONI

WHODUNIT può anche risultare un gioco semplice, ma siamo certi che non vi stancherà mai.

La sfida tra voi e l'assassino è sempre avvincente, sia che avvenga sullo schermo di un televisore o su un tavolo da gioco.

SPECIALE QL

TUTTO L'HARDWARE

TUTTO IL SOFTWARE

La commercializzazione del QL è stata molto travagliata.

Molti sono stati i rinvii, che hanno spesso spazientito i fedelissimi seguaci di Sir Clive Sinclair, i quali non vedevano l'ora di fare conoscenza del nuovo gioiello.

Intenzione della casa inglese era quello di realizzare un computer dalle caratteristiche altamente professionali, ma, grazie al suo basso costo, che avesse una diffusione di massa.

Ora, dopo circa un anno, il QL stenta a decollare e non è ancora molto il software disponibile per una macchina dalle caratteristiche eccezionali.

In questo speciale vogliamo presentarvi una panoramica su tutto l'hardware e il software disponibile in Italia.

IL COMPUTER

Doveroso iniziare dal protagonista di tutto il sistema, il QUANTUM LEAP, meglio conosciuto come QL. Il computer utilizza il microprocessore Motorola 68008 a 32 bit.

La memoria ROM è di 48 K, mentre la memoria RAM è di 128 K, facilmente espandibile fino a 640 K.

La macchina viene fornita con incorporati due microdrive da 100K ed allegate quattro microcartucce contenenti word processing, data base, foglio elettronico e un programma di grafica commerciale.

Il linguaggio utilizzato è il SUPERBASIC, un ulteriore sviluppo dell'ormai collaudato BASIC Sinclair.

I tre principali vantaggi del SUPERBASIC consistono nella possibilità di essere ampliato, di strutturare l'intero programma, e inoltre la sua velocità di esecuzione non dipende dalle dimensioni del programma. Tre caratteristiche adeguate alla notevole memoria ed indispensabili per la realizzazione di programmi particolarmente lunghi e complessi.

L'altra novità è il QDOS, il nuovo sistema operativo realizzato appositamente per il QL.

Il QDOS è molto potente e, utilizzando come linguaggio sempre il SUPERBASIC, sfrutta come non mai la potenza di un microprocessore a 32 bit.

Una caratteristica interessante è l'uso del multi-

tasking che dà la possibilità ad un singolo utente di elaborare simultaneamente parecchi programmi in linguaggio macchina.

Il QL visualizza i risultati in finestre sullo schermo. Ciascun programma in corso di esecuzione può essere visualizzato indipendentemente, entro la sua finestra. Inoltre il QDOS ha un sistema di input/output completamente indipendente dalle unità collegate.

La grafica è ad alta risoluzione, con una definizione di 512 x 256 pixel a quattro colori e 256 x 256 a otto.

Sono incorporate anche due interfacce RS 232, due porte per il joystick ed è possibile inserire cartucce ROM.

Il prezzo (ultimamente in notevole ribasso N.d.R.) è di lire 1.149.000 IVA esclusa (foto 4).

QL MONITOR 14" a colori

Di colore nero e dal design curato, questo monitor ad alta risoluzione è l'ideale per evidenziare l'eccezionale grafica e i colori stupendi del vostro computer.

Lo schermo è speciale, antiriflesso, mentre le dimensioni sono di mm. 375 di lunghezza per 375 di altezza e 390 di profondità.

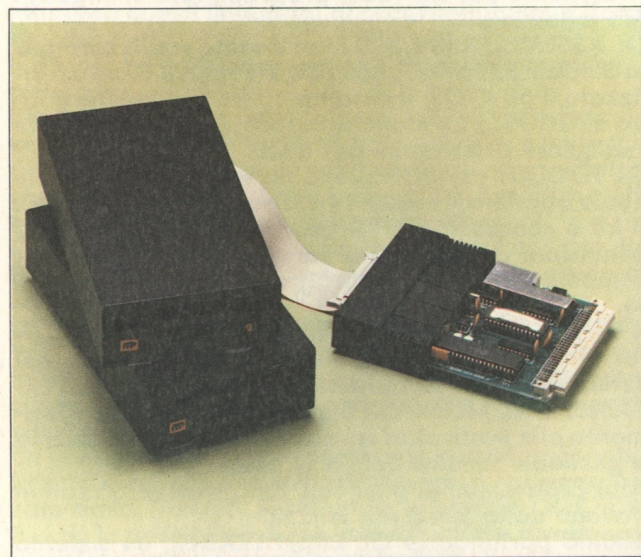
Lo schermo è a 85 colonne, mentre la banda video è di 12 MHz.

Il prezzo è di 600.000 Lire, IVA esclusa (foto 1).

STAMPANTE

La SEIKOSHA è sicuramente la marca produttrice di stampanti più conosciuta in Italia: per il QL propone il modello SP 800 ad impatto, con matrice di stampa 9 x 9.

La stampante viene venduta completa di interfaccia e cavo di collegamento per il computer.



Le caratteristiche di questa periferica sono: 80 colonne, 80 caratteri al secondo, bidirezionale.

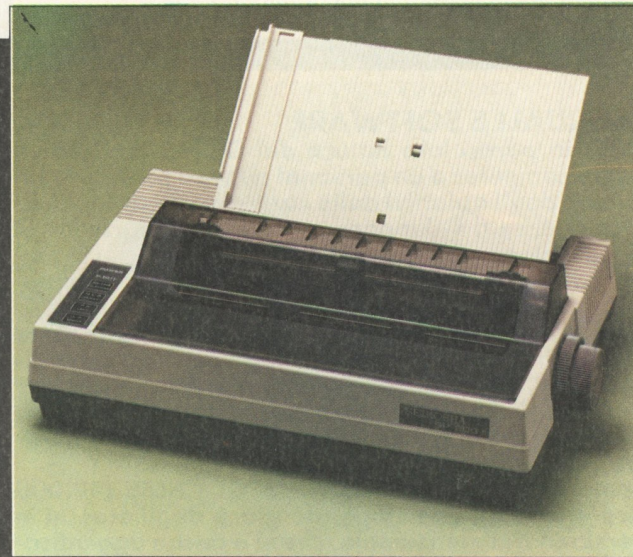
Il foglio può essere continuo o singolo e viene mosso da trattori e frizione.

Il prezzo della SP 800 è di 890.000 Lire, IVA esclusa (foto 2).

UNITÀ DI MASSA

I discussi microdrive forniti di serie possono essere collegati con altri fino ad un massimo di sei.

Il microdrive a 100 K ha una velocità di accesso media di 3-5 secondi e un programma o un file può essere trasferito nella memoria alla velocità di 15



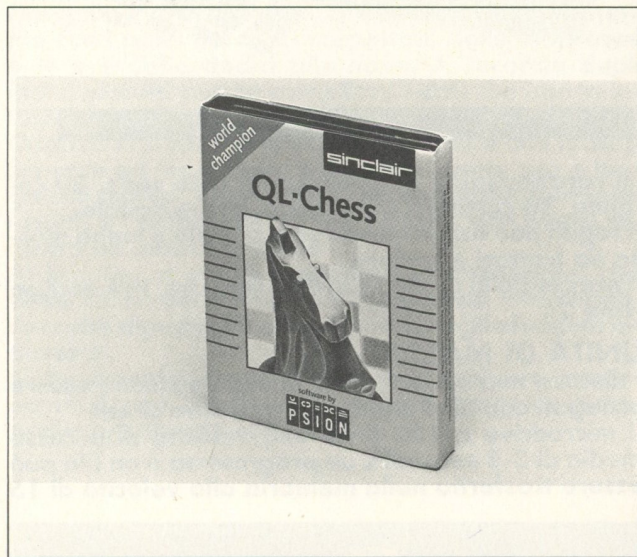
Insert

K per secondo. Ogni microdrive aggiunto costa 169.000 Lire, IVA esclusa.

È prevista anche la realizzazione di un floppy da 5" 1/4, mentre è disponibile un micro floppy drive da 3" 1/2 con una capacità di 720 K al prezzo di 850.000 Lire, IVA esclusa. L'eventuale unità aggiuntiva costa circa la metà (foto 3).

ESPANSIONI

La già discreta memoria di base può essere aumentata, grazie a quattro tipi di espansioni, da 64 fino a 512 K, per un totale di 640 K. Questa eccezionale quantità di memoria consente di utilizzare potenti programmi e di sfruttare al massimo le elevate prestazioni del potente microprocessore. Il prezzo varia, in relazione alla capacità, da 299.000 a 899.000 Lire, IVA esclusa (foto 5).



MANUALI E SOFTWARE

Vista la particolare natura del QL, a metà tra un home computer e un personal per uso professionale, uno degli obiettivi delle case di software è quello di riuscire a sviluppare soprattutto programmi di applicazione professionale.

Recentemente è stata realizzata una nuova versione, denominata 2.0, dei quattro programmi realizzati dalla Psion e forniti con il QL nella confezione di vendita. Oltre ad avere maggiore memoria disponibile, sono state aggiunte alcune opzioni per la scelta della stampante e per la definizione della configurazione del sistema, sia in termini di hardware che di software.

Sempre la Psion ha realizzato QL CHESS (49.000 Lire - foto 8), uno stupendo gioco degli scacchi in versione tridimensionale, che si è anche aggiudicato il campionato mondiale di scacchi per microcom-



puter. Oltre alla stupenda grafica ad alta risoluzione, questo programma prevede 28 livelli di gioco, più di 4000 mosse di apertura e varie funzioni di apprendimento e di aiuto in caso di errore.

La TALENT ha realizzato un package grafico contenente finora un programma per disegnare, a cui se ne aggiungeranno altri.

Di GRAPHICS QL parliamo più approfonditamente in altra parte dell'inserto.

Per chi programma, e vuole conoscere fino in fondo le qualità del suo computer, la Sinclair ha realizzato alcuni programmi e manuali.

TECHNICAL GUIDE (90.000 Lire) è l'utile guida di riferimento per chi vuole programmare in linguaggio macchina.

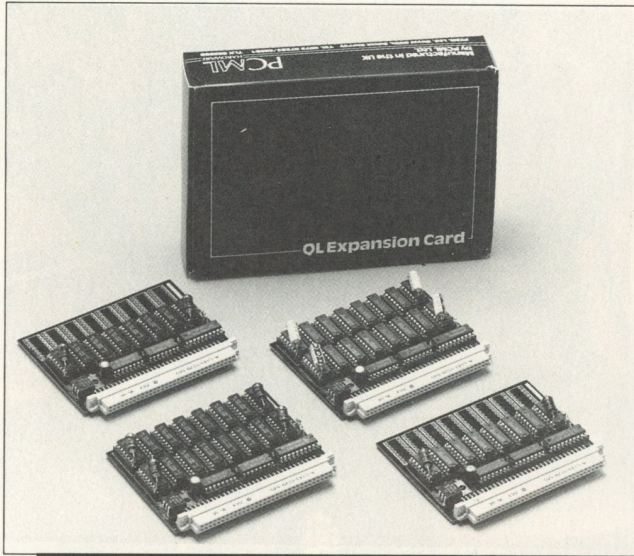
QL ASSEMBLER (89.000 Lire) è stato realizzato per la Sinclair dalla GST, la software house che ha realizzato il 68 K/OS, il sistema operativo in alternativa al QDOS. È un aiuto indispensabile per chi vuole realizzare programmi per il QL.

QL TOOLKIT (59.000 Lire) è un valido aiuto per chi vuole ottenere il massimo dal potente SUPERBASIC. Il kit è composto da 19 programmi di utility e 56 estensioni di linguaggio ed uno screen editor per compilare e cambiare programmi più velocemente.

Il QL MONITOR (59.000 Lire) comprende un monitor di codice macchina, un disassembler ed un assembler completamente integrato con il SUPERBASIC. Il QL MONITOR è stato progettato in riferimento alle particolari caratteristiche di multitasking e gestione finestra del QL (foto 6).

Altri programmi di gestione sono stati realizzati in italiano dalla SINCLAIR ITALIA.

Stiamo parlando di QL CASH TRADER, un sistema di contabilità, che permette di gestire flussi di cas-



La MCC ha realizzato un altro assembler, **ASSEMBLER DEV. KIT** (89.000 Lire), il **QL LISP DEV. KIT** (139.000 Lire), per sfruttare al meglio il linguaggio Lisp e il **QL BCPL DEV. KIT** (139.000 Lire), particolarmente indicato per sviluppare utility, giochi e pacchetti applicativi (foto 10).

Completano questa veloce carrellata due titoli della GST.

QL "C" compiler (169.000 Lire) per l'applicazione del linguaggio "C", usato nei sistemi UNIX, e il **68 K/OS** (310.000 Lire), una scheda da inserire nello slot di espansione che permette l'utilizzo con possibilità di commutare, in alternativa, il già citato **68 K/OS** con l'originale QDOS. Il kit comprende utility, editor e mantenimento file (foto 11).

Infine sono disponibili programmi per il QL, un libro realizzato da **GAETANO MARANO** per il Gruppo Editoriale Jackson che viene venduto con allegata una microcassetta.

sa, bilancio di entrate uscite e situazione IVA; **QL PROJECT PLANNER** (89.000 Lire), che vi permette di rendere graficamente sul monitor un progetto secondo la tecnica Pert; **QL DECISION MAKER** (89.000 Lire), un programma che vi aiuta nella pianificazione degli investimenti e **QL ENTREPRENEUR** (89.000 Lire), un'applicazione utile, sempre in campo commerciale, con la possibilità di simulare il flusso degli affari per 18 mesi successivi.

La Computer One ha realizzato **ASSEMBLER** (79.000 Lire), **PASCAL** (99.000 Lire), **MONITOR** (59.000 Lire) e **FORTH** (99.000 Lire), quattro programmi studiati per sfruttare ed approfondire al massimo questi linguaggi di programmazione (foto 9).

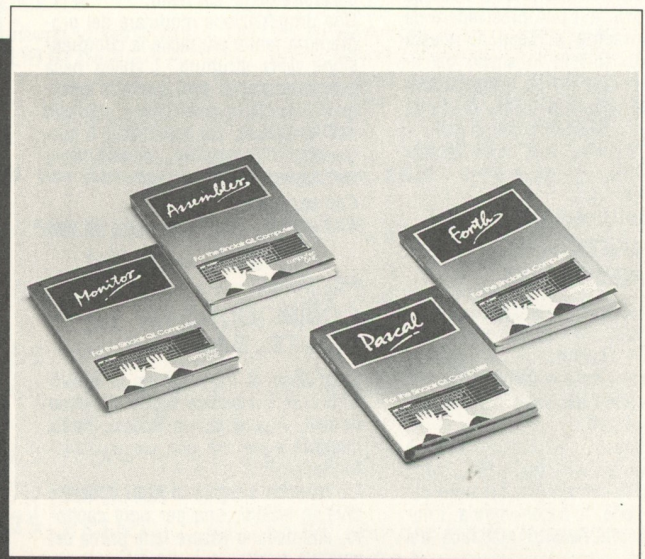
Iniziano ad essere presentati al pubblico anche i primi videogiochi per il QL.

Realizzato per la Eidersoft da Janko Mrfic Flogel, un ragazzo jugoslavo di 16 anni, **ZAPPER** è un arcade che sfrutta l'eccezionale grafica con un effetto simile a Galaxians. Sono previsti 11 livelli, con 18 alieni contemporaneamente sullo schermo.

Il giovane jugoslavo ha anche realizzato, per la Sinclair, **QL CAVERNS**, un platform game simile a Jet Set Willy, con più di 50 locazioni.

La Metacomco ha invece ultimato il **QL PASCAL DEVELOPMENT KIT**, un package contenente anche una ROM con il sistema operativo, due microcartucce che contengono editore, compilatore e l'immanicabile **REFERENCE GUIDE**.

Un utilissimo package, contenente 5 routine per utilizzare al meglio il vostro microdrive, è stato realizzato dalla TALENT. Si chiama **CARTRIDGE DOCTOR** e vi dà la possibilità di copiare singoli file o cartucce complete, recuperare file cancellati ed intervenire su quelli eventualmente danneggiati.



Novità Jackson.

David Lawrence
**LINGUAGGIO
MACCHINA DEL
COMMODORE 64**

Il libro apre nuovi orizzonti a tutti coloro che sono interessati alla programmazione in linguaggio macchina del COMMODORE 64.

Con cassetta
Cod. 572D Pag. 208 Lire 29.000

Clive Prigmore
**IL BASIC IN 30 ORE
PER SPECTRUM**

Questo semplice corso di autoistruzione insegna a programmare, e un programma ha sempre bisogno di due ingredienti, un linguaggio e una struttura: dunque questo libro non insegna solo il BASIC, ma anche come si organizza correttamente un buon programma.

Cod. 501B Pag. 360 Lire 40.000

Rodnay Zaks
**IL TUO PRIMO
PROGRAMMA IN BASIC**

La diffusione del BASIC per la sua semplicità e quasi "naturalità" di programmazione fa sì che una cultura generale sull'informatica e la sua applicazione non può prescindere da una conoscenza di base di questo linguaggio. Questo lo scopo del libro: permettere anche a chi ha soltanto una cultura di base, di capire che cos'è il BASIC e come si usa.

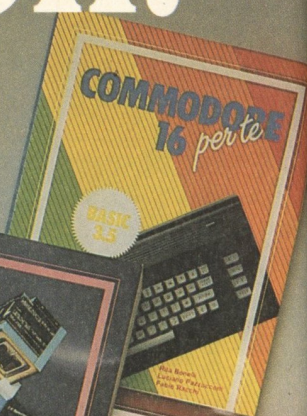
Cod. 507B Pag. 216 Lire 19.500

Czes Kosniowski
**MATEMATICA E
COMMODORE 64**

Tutte le funzioni matematiche disponibili sul C64 sono qui descritte, ed il loro uso è illustrato con programmi che possono essere utilizzati dal lettore all'interno dei suoi, per particolari applicazioni.

Il libro contiene anche informazioni e programmi su altri argomenti, come i codici e la crittografia, i numeri casuali, le serie, la trigonometria, i numeri primi e l'analisi statistica dei dati.

Con cassetta
Cod. 570D Pag. 160 Lire 24.000



F. Williams
**AI CONFINI
DELLO SPECTRUM
Applicazioni avanzate**

Un esame attento dei listati consentirà al lettore di apprendere i "segreti" della programmazione strutturata e migliorare notevolmente le proprie capacità di programmatore.

I programmi presentati vanno dagli arcade più famosi, tra cui il celebre "Spectrum Invaders" ai programmi di utilità più interessanti, dai giochi d'azzardo ai programmi didattici, dai programmi funzionali a quelli di matematica e di giochi di strategia.

Con cassetta
Cod. 414B Pag. 180 Lire 28.000

Mike Grace
**ADVENTURE E
COMMODORE 64**

Un manuale per ideare e utilizzare programmi di Adventure basati esclusivamente sul testo.

Una progettazione modulare del programma rende più facile la comprensione della struttura. L'abilità nella programmazione avrà modo di emergere durante la costruzione graduale dell'Adventure, per modificare il programma dimostrativo, o addirittura per costruirne uno nuovo.

Con cassetta
Cod. 571D Pag. 240 Lire 35.000

Rita Bonelli
**COMMODORE 16
PER TE: BASIC 3.5**

È un libro di introduzione al BASIC C16 con il classico taglio didattico Bonelli. Adatto per la Scuola media inferiore e per chi non conosce l'informatica.

La cassetta allegata al libro contiene diverse lezioni, una per ogni capitolo, che devono essere lette prima del capitolo relativo.

Con cassetta
Cod. 413B Pag. 296 Lire 35.000



**GRUPPO
EDITORIALE
JACKSON**

La biblioteca che fa testo.

ritagliare (o fotocopiare) e spedire in busta chiusa a:
GRUPPO EDITORIALE JACKSON - Divisione Libri - Via Rosellini, 12 - 20124 Milano

CEDOLA DI COMMISSIONE LIBRARIA

VOGLIATE SPEDIRMI

n° copie	codice	Prezzo unitario	Prezzo totale
Totale			

Pagherò contrassegno al postino il prezzo indicato più L. 3.000 per contributo fisso spese di spedizione.

Condizioni di pagamento con esenzione del contributo spese di spedizione:

Allego assegno della Banca

Allego fotocopia del versamento su c/c n. 11666203 a voi intestato

n° _____

Allego fotocopia di versamento su vaglia postale a voi intestato

Nome _____

Cognome _____

Via _____

Cap _____

Città _____

Prov. _____

Data _____

Firma _____

Spazio riservato alle Aziende. Si richiede l'emissione di fattura

**ORDINE
MINIMO
L. 50.000**

Partita I.V.A. _____

ZX CLUB

Il club
Sinclair

Dopo i club Sinclair esclusivamente dedicati allo Spectrum, il tema dominante di questo periodo è l'entrata in scena dei "QL user club". Il QL sta guadagnando gradualmente una fiducia tutto sommato meritata, ed è molto importante, data la relativa scarsa diffusione dei programmi disponibili, che gli amanti dei 32 bit sappiano di potersi appoggiare a qualcuno in caso di difficoltà. Abbiamo già da tempo notizia dell'esistenza di alcuni club operanti con competenza e spirito di iniziativa e di cui contiamo di pubblicare nei prossimi numeri qualche contributo, senza naturalmente dimenticare lo Spectrum, che regnerà incontrastato ancora molto a lungo... E ora andiamo a incominciare!

Gli annunci di ZX Club

QL Sinclair: vendo programmi originali con manuali, linguaggi, giochi, programmi gestionali, libri e traduzioni articoli. Per informazioni scrivete, aggiungendo un francobollo per la risposta, a

QL USER CLUB
via Duccio 3
50047 PRATO FI

IL GRUPPO UTILIZZATORI COMPUTER SINCLAIR NAPOLI ricerca in tutta Italia ditte disposte a praticare agevolazioni e sconti ai soci del club, sul proprio materiale in vendita. Si assicura adeguata pubblicità sul (l'eccezionale. N.d.R.) bollettino interno del club ad ampia diffusione. Indirizzare offerte a

GRUPPO UTILIZZATORI COMPUTER SINCLAIR NAPOLI
c/o Roberto CHIMENTI
via Luigi Rizzo 18
80124 NAPOLI NA

tel. 081/617368
081/7623121

I due numeri di telefono riportati non sono entrambi funzionanti: nel momento in cui scriviamo il primo è provvisorio, ma ancora in uso, mentre il secondo è quello definitivo, ma non operativo. Per chi legge e fosse interessato l'unica possibilità è provare!

I programmi di ZX CLUB

Prosegue in questo numero la serie dei test del SINCLAIR CLUB ROMA - TA-

LENTI: l'argomento del mese è questa volta la sincerità. "Molti di noi" scrive l'autore del programma "ritengono di essere sinceri, anche se qualche volta sono costretti dagli avvenimenti e dall'ambiente in cui vivono a non dire sempre la verità nuda e cruda. Tutto ciò dipende dallo spirito di adattamento individuale e dai compromessi che si sanno impostare. Tuttavia ciò non vuol dire che siamo ipocriti e falsi, a meno che il numero delle menzogne sia così elevato da costituire il nostro normale comportamento".

Al test possono prendere parte più persone: basta inserirne il numero complessivo e i nomi. Per rispondere non dovrete fare altro che scegliere tra i tasti "A", "B" e "C".

Ma... attenzione! Il test ci dirà se siamo sinceri: ma come faremo a sapere se il test è sincero? Mah...nel caso il dubbio vi perseguiti provate a scrivere al

SINCLAIR CLUB ROMA - TALENTI
c/o Massimo D'ASCENZO
via F. D'Ovidio 109
00175 ROMA RM
tel. 06/8280043

Test: siete sinceri?

di Massimo D'Ascenzo - Sinclair Club Roma - Talenti versione per ZX Spectrum 16K - introduzione

```
5 BORDER 0: PAPER 0: INK 6:  
  BRIGHT 1: CLS  
10 PRINT AT 1,1; INK 5;"T E S T :"  
15 PRINT INK 5;" Siete sinceri? "  
20 LET z=0: LET t=10  
25 LET j=3.73: LET gr=66.91  
30 LET xx=127-t/2: LET yy=87-t/2:  
  PLOT xx,yy  
35 LET x=t*SIN z: LET y=t*COS z:  
  GO SUB 060  
40 LET x=t*COS z: LET y=-t*SIN z:  
  GO SUB 060
```

```
45 LET x=-t*SIN z: LET y=-t*COS z:  
  GO SUB 060  
50 LET x=-t*COS z: LET y=t*SIN z:  
  GO SUB 060  
55 GO TO 035  
60 IF x+PEEK 23677>200 THEN  
  GO TO 075  
65 DRAW x,y: LET z=z-gr: LET t=t+j  
70 RETURN  
75 PRINT INK 5;AT 20,1;"by Massimo  
  e Cinzia D'ASCENZO"
```

```

80 PRINT INK 6;" del Sinclair Club
   Roma-Talenti"
85 PRINT AT 8,23; INK 3;"TEST":
   PRINT AT 17,2; INK 7;"TEST"
90 INK 0: LOAD ""

```

Test: siete sinceri?
programma principale

```

0 REM by Massimo e Cinzia D'ASCENZ
0
100 LET p=1: BRIGHT 0: LET m=0:
   LET n=0: LET r=0: GO SUB 1990
110 POKE 23658,8: REM caratteri maiu
   scoli
120 PRINT AT 4,11;"T E S T"
130 PRINT AT 1,0;"{SG8}{G8}{SG8}
   {G8}{SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}"
140 PRINT AT 3,0;"{SG8}{G8}{SG8}
   {G8}{SG8}{G8}{SG8}{G8}{SG8}";
   AT 3,22;"{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}"
150 PRINT AT 4,0;"{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}";
   AT 4,22;"{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}"
160 PRINT AT 5,0;"{SG8}{G8}{SG8}
   {G8}{SG8}{G8}{SG8}{G8}{SG8}";
   AT 5,22;"{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}"
170 PRINT AT 6,0;"{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}";
   AT 6,22;"{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}"
180 PRINT AT 7,0;"{SG8}{G8}{SG8}
   {G8}{SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}{SG8}{G8}
   {SG8}{G8}{SG8}{G8}"
190 PRINT AT 14,8;"Siete sinceri?"
200 PRINT AT 15,5;"
   "
210 INPUT "N. giocatori? ";g
220 DIM K$(g,9): REM K$=Nome dei gio
   catori
230 FOR y=1 TO g
240 INPUT "Nome giocatore ";(y);" ?
   ";K$(y)

```

```

250 NEXT y
260 DIM A$(15*g): REM 15*g=N.domande
   * N. giocatori
270 GO SUB 2020
280 GO SUB 2380
290 PRINT AT 2,0; OVER 1;"E' vero ch
   e ogni tanto pensate acose cosi'
   brutte da non poterne parlare ?"
300 PRINT AT 8,0;"a) Mai"
310 PRINT
320 PRINT "b) Molto raramente"
330 PRINT
340 PRINT "c) Qualche volta"
350 GO SUB 2380
360 PRINT AT 2,0; OVER 1;"E' vero ch
   e ogni tanto vi scappa qualche pa
   rolaccia, imprecazione o insulto
   ?"
370 PRINT AT 8,0;"a) Non e' vero, mi
   controllo sempre"
380 PRINT
390 PRINT "b) Quasi mai"
400 PRINT
410 PRINT "c) Si', ogni tanto"
420 GO SUB 2380
430 PRINT AT 2,0; OVER 1;"E' vero ch
   e talora vi capita di raccontare
   qualche frottola ?"
440 PRINT AT 8,0;"a) Quasi mai"
450 PRINT
460 PRINT "b) Non racconto mai frott
   ole"
470 PRINT
480 PRINT "c) Una, ma molto rarament
   e"
490 GO SUB 2380
500 PRINT AT 2,0; OVER 1;"E' vero ch
   e non sempre fate quello che
   vi siete proposto ?"
510 PRINT AT 8,0;"a) No, faccio semp
   re quello che mi sono propost
   o"
520 PRINT
530 PRINT "b) Praticamente posso dir
   e di fare sempre cio' che m
   i propongo"
540 PRINT
550 PRINT "c) Si, talvolta non lo fa
   ccio"
560 GO SUB 2380
570 PRINT AT 2,0; OVER 1;"Qualche vo
   lta vi arrabbiate ?"
580 PRINT AT 8,0;"a) No, non mi arra
   bbio mai"
590 PRINT
600 PRINT "b) Mi arrabbio solo se mi
   fanno perdere la pazienza"
610 PRINT
620 PRINT "c) Effettivamente qualche
   volta mi arrabbio anche se n
   on c'e' un motivo grave"

```

```

630 GO SUB 2380
640 PRINT AT 2,0; OVER 1;"Talvolta r
    imandate a domani cio'che potres
    te fare oggi ?"
650 PRINT AT 8,0;"a) No, non rimando
    mai, anzi"
660 PRINT
670 PRINT "b) Mi sforzo di fare quel
    lo che devo al momento opport
    uno"
680 PRINT
690 PRINT "c) Si', qualche volta mi
    capita"
700 GO SUB 2380
710 PRINT AT 2,0; OVER 1;"E' vero ch
    e ci sono dei momenti in cui sie
    te, senza motivo, di cattivo
    umore ?"
720 PRINT AT 8,0;"a) Non sono mai di
    cattivo umore senza motivo"
730 PRINT
740 PRINT "b) Molto raramente"
750 PRINT
760 PRINT "c) Qualche volta"
770 GO SUB 2380
780 PRINT AT 2,0; OVER 1;"E' vero ch
    e a tavola vi comportate
    peggio in casa che fuori in c
    ompagnia?"
790 PRINT AT 8,0;"a) No, non e' vero
    "
800 PRINT
810 PRINT "b) Quasi mai"
820 PRINT
830 PRINT "c) Talvolta"
840 GO SUB 2380
850 PRINT AT 2,0; OVER 1;"Se poteste
    entrare allo stadio oal cinema
    o sul treno senza pagare il
    biglietto, perche' sicuri di
    non essere scoperti, lo fareste
    ?"
860 PRINT AT 8,0;"a) No, per princip
    io"
870 PRINT
880 PRINT "b) Di solito non faccio q
    ueste cose"
890 PRINT
900 PRINT "c) Forse, se proprio fossi
    sicuro che nessuno se ne acco
    rge"
910 GO SUB 2380
920 PRINT AT 2,0; OVER 1;"C'e' chi d
    ice ~Preferisco perdere un
    a partita piuttosto che vincer
    la~"
930 PRINT AT 8,0;"a) Effettivamente
    qualcuno puo' avere un motivo
    per dire queste cose"
940 PRINT
950 PRINT "b) In certi casi puo' dar
    si"
960 PRINT
970 PRINT "c) Ritengo che nessuno po
    ssa dire una cosa cosi' st
    upida"
980 GO SUB 2380
990 PRINT AT 2,0; OVER 1;"Cercate di
    conoscere persone importanti
    per sentirvi importanti
    ?"
1000 PRINT AT 8,0;"a) Non mi e' mai c
    apitato"
1010 PRINT
1020 PRINT "b) No, non e' possibile,
    sono stupidaggini"
1030 PRINT
1040 PRINT "c) Qualche volta"
1050 GO SUB 2380
1060 PRINT AT 2,0; OVER 1;"Tutte le p
    ersone che conoscete vi sono si
    mpatiche ?"
1070 PRINT AT 8,0;"a) certamente"
1080 PRINT
1090 PRINT "b) Si', perche' no"
1100 PRINT
1110 PRINT "c) No, molte no"
1120 GO SUB 2380
1130 PRINT AT 2,0; OVER 1;"Qualche vo
    lta fate dei pettegolez
    zi ?"
1140 PRINT AT 8,0;"a) Non mi e' mai c
    apitato"
1150 PRINT
1160 PRINT "b) Penso solo ai fatti mi
    ei e non mi preoccupo di qu
    elli degli altri"
1170 PRINT
1180 PRINT "c) Si, spesso"
1190 GO SUB 2380
1200 PRINT AT 2,0; OVER 1;"Vi diverti
    te alle barzellette sporche ?"
1210 PRINT AT 8,0;"a) No, le barzelle
    tte sporche non mi diverton
    o"
1220 PRINT
1230 PRINT "b) Molto raramente"
1240 PRINT
1250 PRINT "c) Si, solo qualche volta
    "
1260 GO SUB 2380
1270 PRINT AT 2,0; OVER 1;"Vi capita
    di dare giudizi su persone ch
    e non conoscete ?"
1280 PRINT AT 8,0;"a) No, non c'e' mo
    tivo"
1290 PRINT
1300 PRINT "b) Non mi e' mai capitato
    "
1310 PRINT
1320 PRINT "c) Si, qualche volta mi e

```

```

      '          capitato"
1330 GO SUB 1840
1340 DATA "Siete assolutamente normali,"
1350 DATA "capaci di adattarvi"
1360 DATA "destreggiandovi tra le"
1370 DATA "difficolta' della vita. Accettate"
1380 DATA "il compromesso, ma solo se"
1390 DATA "non e' troppo pesante e non vi"
1400 DATA "impone di rinunciare ai vostri"
1410 DATA "principi. La vostra opinione e'"
1420 DATA "che qualche volta le menzogne"
1430 DATA "pietose occorrono, pur di non far"
1440 DATA "del male a un vostro caro."
1450 DATA "PREMI UN TASTO PER CONTINUARE"
1460 DATA "Siete troppo sinceri e questo vi"
1470 DATA "creera' non pochi grattacapi."
1480 DATA "Siete tutti di un pezzo, non"
1490 DATA "guardate in faccia a nessuno e"
1500 DATA "dite a ognuno cio' che merita,"
1510 DATA "ma la vostra onesta' e la vostra"
1520 DATA "sincerita' vi procureranno piu' "
1530 DATA "nemici che amici."
1540 DATA "PREMI UN TASTO PER CONTINUARE"
1550 DATA "Siete amanti delle frodole,"
1560 DATA "tenete a presentarvi non come"
1570 DATA "realmente siete, ma come pensate"
1580 DATA "che gli altri vi desiderino."

1590 DATA "Rispondete all'ambiente in modo"
1600 DATA "da dar l'impressione di essere"
1610 DATA "retti, onesti, attaccati a i"
1620 DATA "principi morali, mentre forse"
1630 DATA "non lo siete tanto."
1640 DATA "PREMI UN TASTO PER CONTINUARE"

1650 GO SUB 1990: PRINT AT 8,10;"UN MOMENTO..."
1660 PAUSE 50
1670 PRINT AT 16,2;"...STO ELABORANDO I RISULTATI"
1680 DIM f(g*15)
1690 FOR e=1 TO g*15
1700 IF A$(e)="A" THEN LET f(e)=1
1710 IF A$(e)="B" THEN LET f(e)=1
1720 IF A$(e)="C" THEN LET f(e)=0
1730 NEXT e
1750 DIM h(g): REM h=punti giocatori
1760 FOR l=1 TO g
1770 LET h(l)=0
1780 LET z=0: GO TO 1800
1790 LET z=z+1
1800 LET h(l)=h(l)+f(g*z+1)
1810 IF z<14 THEN GO TO 1790
1820 NEXT l
1830 GO TO 1920
1840 LET n=n+1
1850 LET r=r+1
1860 PRINT AT 20,3;K$(r)
1870 PRINT AT 20,18;" ": GO SUB 2470
1880 LET A$(n)=INKEY$
1890 BEEP .02,23: PAUSE 5: BEEP .3,19

1900 IF r=g THEN LET r=0: GO SUB 2420: RETURN
1910 GO TO 1840
1920 CLS
1930 FOR l=1 TO g
1940 IF h(l)<=2 THEN GO SUB 1990: PRINT INVERSE l;K$(l),"Punti ";h(l): GO SUB 2200
1950 IF h(l)>=3 AND h(l)<=11 THEN GO SUB 1990: PRINT INVERSE l;K$(l),"Punti ";h(l): GO SUB 2180
1960 IF h(l)>11 THEN GO SUB 1990: PRINT INVERSE l;K$(l),"Punti ";h(l): GO SUB 2220
1970 NEXT l
1980 GO TO 2030
1990 LET p=p+1
2000 IF p=5 THEN LET p=0
2010 PAPER p: BORDER p: INK 8: CLS : RETURN
2020 PRINT AT 21,0;"PREMI UN TASTO PER CONTINUARE": PAUSE 0: RETURN
2030 GO SUB 1990: PRINT AT 7,9;"Vuoi rifare": PRINT AT 15,7;"il TEST (S/N) ?"
2040 IF INKEY$="" THEN GO TO 2040
2050 CLS : IF INKEY$="S" THEN GO TO 100
2060 DATA "Il SINCLAIR CLUB ROMA-TALENTI"
2070 DATA "nella speranza che questo TEST"
2080 DATA "ti sia piaciuto, ti invita

```

```

a"
2090 DATA "comunicarci le tue impress
ioni,"
2100 DATA "positive o negative scrive
ndo a"
2110 DATA "SINCLAIR CLUB ROMA-TALENTI
"
2120 DATA "c/o D'ASCENZO MASSIMO"
2130 DATA "Via F. D'Ovidio 109 - ROMA
"
2140 DATA "Tel. 06/828.00.43"
2150 RESTORE 2060
2160 LET s=9: GO SUB 2240
2170 STOP
2180 RESTORE 1340
2190 LET s=12: GO TO 2240
2200 RESTORE 1460
2210 LET s=9: GO TO 2240
2220 RESTORE 1550
2230 LET s=10: GO TO 2240
2240 FOR t=1 TO s
2250 READ I$
2260 PRINT
2270 PRINT
2280 POKE 23692,255
2290 FOR u=1 TO LEN I$
2300 IF CODE I$(u)=32 THEN GO TO 232
0
2310 BEEP .01,1
2320 PRINT I$(u);
2330 NEXT u
2340 BEEP .08,25
2350 NEXT t
2360 PAUSE 0
2370 CLS : RETURN
2380 LET m=m+1
2400 IF m>1 THEN GO SUB 1840
2405 GO SUB 2590
2410 GO SUB 1990: PRINT INVERSE 1;"D
OMANDA N.": PRINT AT 0,10;
INVERSE 1;m: GO TO 2430
2420 FOR i=1 TO 21: RANDOMIZE USR 328
0: NEXT i: RETURN
2430 FOR q=1 TO 6
2440 PRINT INK 7;AT q,0;"{32SG8}"
2450 NEXT q: INK 7: RETURN
2460 STOP
2470 OVER 0: LET J$="Quale risposta s
cegli: A - B - C "
2480 LET H$=" "
2490 LET B$=J$+H$
2500 LET C$=B$
2510 PRINT AT 21,0; INK 7;C$(1 TO 32)

2520 LET C$=C$(2 TO )
2530 IF LEN C$<32 THEN LET C$=C$+B$
2540 PAUSE 10
2550 IF INKEY$="A" OR INKEY$="B" OR
INKEY$="C" THEN GO TO 2570
2560 GO TO 2510
2570 PRINT AT 20,18; INK 7;;INKEY$

```

```

2580 RETURN
2590 CLS : PRINT AT 4,8; INVERSE 1;"D
O M A N D A": LET y$=STR$ m
2600 IF m<=9 THEN GO TO 2640
2610 LET w$=y$( TO 1)
2620 IF m>9 THEN LET c=m-10
2630 LET y$=STR$ c
2640 LET d=256+256*PEEK 23607+PEEK 23
606
2650 LET a=0
2660 LET a=a+1
2670 LET y=CODE y$
2680 IF m<10 THEN LET w$="0"
2690 LET w=CODE w$
2700 FOR j=0 TO 7
2710 LET byte=PEEK (d+j+(y-32)*8)
2720 LET b=PEEK (d+j+(w-32)*8)
2730 FOR k=1 TO 8
2740 LET x=INT (byte/2): LET bit=byte
-2*x: LET byte=x
2750 LET X=INT (b/2): LET bt=b-2*X:
LET b=X
2760 IF bit=0 THEN GO TO 2780
2770 PRINT AT j+7,25-k;CHR$ (143)
2780 IF bt=0 THEN GO TO 2800
2790 PRINT AT j+7,14-k;CHR$ (143)
2800 NEXT k
2810 NEXT j
2820 RETURN

```

ERRATA CORRIGE

Il prezzo della tastiera SEGA EMPEROR non è, come erroneamente annunciato nel numero 8 di Supersinc, di L. 1.050.000, ma di L. 100.000 (Iva esclusa).

Ci scusiamo del spiacevole equivoco con i lettori e la ATW studio, distributrice della tastiera.

P R I N T

Recensioni Libri

di **Lucio Bragagnolo**

Il manuale del BASIC – per TI99/4A, VIC 20, C64, ZX81, ZX Spectrum, MZ700, IBM PC, M20, Apple

di **Martino Sangiorgio**
ed. **Gruppo Editoriale Jackson**

Dopo il "Dizionario del BASIC", recensito su queste pagine mesi or sono, prendiamo in esame un altro testo per aspiranti convertitori di programmi e, più in genericamente, sostenitori della necessità di disporre di una cultura generale anche in campo informatico, se non altro per evitare problemi: può sempre capitare infatti di trovarsi prima o poi costretti a smanettare sul computer dell'amico; e che dire di quando si decide di compiere un "salto di qualità" (in inglese Quantum Leap: vi dice niente?), comprando un calcolatore di caratteristiche superiori e ovviamente dotato di un BASIC diverso da quello a cui ci si è abituati? Come per le lingue estere, anche in campo informatico il possedere un'infarinatura di dialetti BASIC può rivelarsi utile e può aumentare le proprie conoscenze, oltre che, soprattutto, la propria capacità di orizzontarsi da soli in uno dei mercati più confusi e infidi che esistano.

Questo libro porta a un confronto tra diverse macchine, raffrontando tra loro i vari dialetti BASIC in esse contenuti: si va dal BASIC serio, "affaristico" e monumentale di M20 e PC IBM a quelli di impostazione vagamente microsoftia-

na, rintracciabili sulla famiglia Commodore e sull'Apple. Abbiamo anche lo Sharp MZ700, che conobbe un periodo di buona diffusione non più tardi di un paio di anni fa, specie negli ambienti universitari. Compiono infine il BASIC Sinclair (ZX81 compreso) e quello del TI99/4A, l'uno e l'altro piuttosto atipici e, comunque, di ben diverso successo sul mercato. A voler essere pignoli ci si potrebbe lamentare per il non eccessivo aggiornamento del libro: mancano, tanto per fare qualche nome, QL, MSX e Macintosh; tuttavia si tratta di prodotti non ancora molto diffusi (MSX) o lontani dalle possibilità e dai bisogni dell'hobbista medio (Mac). Oltretutto il libro avrebbe assunto dimensioni e costi assolutamente improponibili (così siamo già vicini alle 500 pagine...).

L'analisi viene condotta argomento per argomento, ad esempio la cancellazione del video o la gestione delle matrici, presentando i differenti formati delle istruzioni debitamente commentati e corredati da esempi. L'insieme risulta scorrevole da leggere e facile da capire, purché si sia stati sufficientemente accorti da prestare la massima attenzione all'introduzione, dove vengono presentate le notazioni usate nel corso del libro per illustrare il formato delle istruzioni, unitamente alle cosiddette carte sintattiche, ovvero diagrammi che spiegano carattere per carattere come vengono composti i vari comandi; le carte sono estremamente chiarificatrici, a patto però che sia stato ben capito il loro funzionamento.

Il linguaggio usato dall'autore (personaggio di prestigio nel campo dell'elaborazione dati) è estremamente rigoro-

so, ma sempre comprensibile (senza i contorcimenti sintattici tanto cari a certi accademici) e dotato di ottima sintesi. Gli argomenti trattati sono praticamente tutti quelli del BASIC, compresi temi come la generazione del colore e del suono che normalmente finiscono, per la grande diversità di possibilità e di istruzioni a disposizione di ogni macchina, per essere esclusi da trattazioni globali come questa.

Il livello di competenza dell'autore sui singoli computer trattati ci è sembrato buono, probabilmente acquisito dopo una certa pratica e sicuramente non solo derivante dalla semplice lettura dei manuali (che sarebbe stato il modo più facile per scrivere un libro del genere, a prezzo però di un grave scadimento della qualità del prodotto). Ovviamente, è però inutile cercare sul libro i trucchi e le peculiarità facenti parte del bagaglio di un qualunque sinclairista, commodoriano o appl... (non vi si intreccia la lingua?) minimamente esperto, anche perché ciò esulerebbe dal tema principale del libro, cioè il confronto dei dialetti BASIC, sganciati dall'hardware dei rispettivi computer e puliti di bug, stranezze e amenità varie.

Insomma, chi crede ancora che TRON sia solo un film di fantascienza e PAI il nome di una nota marca di patatine trarrà enorme giovamento dalla lettura di questo "Manuale del BASIC": i più smaliziati potranno divertirsi a cercare le differenze tra NUMBER, RENUMBER e RESEQUENCE; e anche quelli che pensano di sapere tutto potranno trovare una quantità di notizie interessanti e utili sul loro computer... e su quello dell'amico.

PRINT

Lo Spectrum a scuola di **Eric Deeson** ed. **Gruppo Editoriale Jackson**

Dite la verità: non sono un po' tutti uguali quei libri pieni di nozioni su come imparare a programmare? A parte le lodevoli eccezioni, non vi annoiano un pochino le solite trattazioni su algoritmi, subroutine, registratori a cassette e collegamenti al TV di casa, ormai lette e rilette fino alla nausea?

Forse quello che vi occorre è un libro sempre dedicato agli appassionati di informatica, ma che sia stimolante, scritto con un linguaggio diverso da quello gelido e rigoroso a cui siamo stati abituati, che possa far nascere dibattiti e discussioni e che, ma sì, ci insegni anche a usare appropriatamente il BASIC Spectrum.

Il libro è questo... e un piazzista potrebbe benissimo terminare qui la sua recensione. In realtà è meglio approfondire un po' l'esame de "Lo Spectrum a scuola", per poter meglio capire di che cosa si tratta.

Consideriamo, per cominciare, la sua destinazione: fin dal titolo è chiaro come questa nuova produzione Jackson sia collegata al mondo della scuola, e più precisamente agli insegnanti, in teoria coloro che dovrebbero condurre i bambini al loro primo contatto con l'informatica e i calcolatori, ma, a volte (pochi) per cattiva volontà e a volte (molte) per problemi finanziari e cecità burocratiche, si trovano ad operare con una dotazione di materiale e di fondi largamente insufficienti. Sotto questo aspetto la lettura de "Lo Spectrum a scuola" sarà estremamente istruttiva: nella presentazione all'edizione italiana, infatti, il traduttore informa di come nel testo siano rimasti "di proposito intatti tutti i riferimenti al sistema scolastico e più in generale alla società britannica", in modo che il lettore possa "constatare il gravissimo ritardo con cui la scuola in Italia si adegua alle nuove richieste dell'ambiente". Ci pare che non vi sia bisogno di commenti.

Tornando più strettamente al libro, esso si divide più o meno in tre sezioni: la prima si incarica di fornire un'introduzione all'uso del computer nella società di oggi, partendo da interrogativi elementari quali "Cos'è un computer?" per finire con una interessante trattazio-

ne del ruolo futuro dei computer nella società e più specificamente nel campo dell'istruzione. La seconda vuole essere un corso di programmazione in BASIC per principianti, diretto agli insegnanti che vogliono imparare a usare lo Spectrum come a quelli che lo conoscono già e vogliono insegnarne l'uso alla classe. L'ultima sezione fornisce una guida di programmazione per lo Spectrum, utile a quei ragazzi che frequentano a scuola corsi di informatica - crediamo, ahinoi, in Italia non moltissimi -, costituita da suggerimenti e proposte di esercizi presenti al termine di ogni capitolo. Completano la struttura generale del libro due appendici, una contenente alcuni semplici programmi "tipo", composti da routine molto usate di randomizzazione, lettura da tastiera, sicure contro gli errori di input e via dicendo, mentre la seconda, intitolata "Consigli", suggerisce alcuni sistemi per semplificare la programmazione dello Spectrum. Chicca da segnalare in questo caso - la sua presenza è dovuta a ragioni particolari, in quanto in tutto il libro il linguaggio macchina è pochissimo trattato - è la presenza di una piccola routine interamente in l/m, che a detta dell'autore impedirebbe allo Spectrum di bloccarsi con un errore quando la funzione VAL viene applicata per sbaglio o per necessità a una stringa non numerica. Il condizionale è d'obbligo, perché non abbiamo verificato la cosa, ma non dovrebbero esserci problemi.

Per quanto riguarda le caratteristiche generali del libro la prima cosa che colpisce è, ne abbiamo già accennato, il linguaggio: brillante, chiaro, qualche volta anche spiritoso e autoironico, sempre però senza perdere di vista lo scopo dell'insegnamento né perdersi in discorsi che non siano strettamente inerenti al tema, sia esso l'uso di PRINT o un'ipotesi futuristica circa i destini dell'informatica (in particolare dell'informatica applicata alla scuola) e della scuola stessa nella nostra società.

I listati sono in genere corti e senza abbellimenti grafici o sonori (a cui dovrà pensare l'utente, lavorandoci sopra secondo le sue esigenze), e sono stampati con una comune Seikosha. Non ci sono problemi di leggibilità; per di più la Jackson allega meritoriamente al libro una cassetta che riporta registrati tutti i programmi presentati, cassetta che non ha presentato problemi di caricamento e, pur presentando un paio di distrazioni del traduttore, consentirà al-

l'utente di risparmiare una gran quantità di tempo.

Anche l'impaginazione è all'altezza: movimentata, ma non troppo, alterna al testo principale alcuni riquadri esplicativi e numerose illustrazioni o foto di carattere, in certi casi, del tutto inedito. Sono presenti anche vignette eseguite dallo stesso disegnatore che cura le edizioni della casa inglese Shiva, di cui sono state presentate in PRINT diverse realizzazioni. Caratterizzate con un umorismo tipicamente inglese alcune colgono nel segno ("Noi non siamo logici: perché dovrebbero esserlo i computer?") e altre, a nostro parere, meno: si tratta comunque di dettagli assolutamente secondari.

Per concludere, abbiamo già formulato un parere positivo su "Lo Spectrum a scuola" già all'inizio di questa recensione: non facciamo quindi altro che riconfermarlo, segnalando anzi particolarmente il libro a insegnanti e scuole, non solo per ciò che riguarda il suo contenuto strettamente programmatico, ma anche per le tematiche in esso accennate, sicuramente meritevoli di attenzione e riflessione.

**I LIBRI ITALIANI
VENGONO
OFFERTI CON IL
10%
DI SCONTO SUL
PREZZO DI COPERTINA**

**I LIBRI POSSONO
ESSERE ORDINATI
UTILIZZANDO IL
COUPON A
PAG. 66**

SUPERSING

CEDOLA DI ORDINAZIONE - CASSETTE
da compilare e spedire in busta chiusa a
J.soft - Viale Restelli, 5 - 20124 Milano - Tel. 6888228

LIBRI

Il manuale del Basic - per T199/4A: VIC20: C64: ZW81: ZW Spectrum:
MZZ700: IBM PC: M20 APPLE L. 40.500
Lo Spectrum a scuola - ed. G.E.J. cod. AJAK041 L. 34.200
cod. AJAC062

Ordino i seguenti libri per un importo totale di L. + L. 2.000
come contributo fisso per spese di spedizione

- Cod. Cod.
- Cod. Cod.
- Contanti allegati Assegno allegato n°
- Ho spedito l'importo a mezzo vaglia postale
- Ho versato l'importo sul ccp n° 19445204 intestato a J.soft - Milano
- Pagherò in contassegno al postino al ricevimento dei volumi (valido solo per i soci in Italia)

Nome

Cognome

Via

Città C.A.P.

Se richiesta fattura:
Cod. F. e P. Iva

Data

Firma

Per i residenti all'estero - pagamento anticipato (vaglia o versamento su ns. ccp)

SUPERSING INPUT/OUTPUT

La rubrica INPUT/OUTPUT è gratuita ed aperta a tutti i lettori. Chi desidera comprare, vendere o cambiare hardware o software può inviare il tagliando a J.soft - Viale Restelli, 5 - 20124 MILANO

- COMPRO VENDO CAMBIO
- ZX80 ZX81 ZX Spectrum QL SOFTWARE
- PERIF.

.....

.....

.....

.....

.....

.....

Nome Cognome

Via C.A.P.

Città Tel.

SUPERSING è bello, però... (ovvero suggerimenti, idee, critiche, richieste e tutto ciò che vi passa per la testa).

.....

.....

.....

.....

.....

.....

Nome Cognome

Via C.A.P.

Città Tel.

i Tascabili



Jackson, naturalmente.

ELENCO DEI TITOLI DISPONIBILI

I tascabili Jackson sono uno strumento prezioso per chi lavora con il computer.

- SINCLAIR SPECTRUM cod. 017H
- VIC 20 cod. 005H
- COMMODORE 64 cod. 002H
- PC IBM cod. 018H
- APPLE IIc cod. 003H
- SHARP MZ80A cod. 014H
- LA PROGRAMMAZIONE cod. 004H
- WORD STAR cod. 008H
- UNIX cod. 009H
- LOGO cod. 020H
- MS-DOS cod. 019H
- PROGRAMMI DI STATISTICA cod. 015H
- CP/M cod. 011H
- PC-DOS cod. 012H

- BASIC cod. 007H
- ASSEMBLER Z80 cod. 016H
- ASSEMBLER 6502 cod. 013H
- COBOL cod. 001H
- FORTRAN 77 cod. 010H
- PASCAL cod. 006H



OGNI TASCABILE COSTA L. 8.500

ritagliare (o fotocopiare) e spedire in busta chiusa a:
GRUPPO EDITORIALE JACKSON - Divisione Libri - Via Rosellini, 12 - 20124 Milano

CEDOLA DI COMMISSIONE LIBRARIA

VOGLIATE SPEDIRMI

n° copie	codice	Titolo	Prezzo unitario	Prezzo totale

Totale

Pagherò contrassegno al postino il prezzo indicato più L. 3.000 per contributo fisso spese di spedizione.

Condizioni di pagamento con esenzione del contributo spese di spedizione:

Allego assegno della Banca Allego fotocopia del versamento sul c/c n. 11666203 a voi intestato

N° _____ Allego fotocopia di versamento su vaglia postale a voi intestato

Nome e Cognome _____

Via _____

Cap _____

Città _____

Prov. _____

Data _____

Firma _____

Spazio riservato alle Aziende Si richiede l'emissione di fattura

ORDINE
MINIMO
L. 50.000

Partita I.V.A. _____

FINALMENTE!

La Softtrivista che ti gasa!



**QUALCOSA DI SUPER, DI INEDITO,
DI IRRESISTIBILE**

**IL VERO GIOCO
COMINCIA ADESSO**

**IN EDICOLA
JACKSON SOFT
SERIE ORO**

I giochi esclusivi per
Commodore 64 e Spectrum 48 K
importati dall'Inghilterra, mai
presentati in Italia.
Una sfida Jackson al già visto, al
già fatto, al... già registrato.



Continua
la serie delle
entusiasmanti
avventure di
WALLY

Corri in edicola, il vero gioco comincia solo adesso
e se sei davvero bravo partecipa alla "sfida al campione",
utilizzando il tagliando che troverai sull'ultima pagina
di copertina di ogni numero.



**GRUPPO
EDITORIALE
JACKSON**